Concurrent Games with Ordered Objectives

Patricia Bouyer, Romain Brenguier,
Nicolas Markey, Michael Ummels

Research report LSV-11-22

Version 2 – January 11, 2012

Laboratoire Spécification & Vérification

École Normale Supérieure de Cachan
61, avenue du Président Wilson
94235 Cachan Cedex France

Concurrent Games with Ordered Objectives

Patricia Bouyer, Romain Brenguier, Nicolas Markey, Michael Ummels

LSV, CNRS & ENS Cachan, France
{bouyer,brenguier,markey,ummels}@lsv.ens-cachan.fr

Abstract. We consider concurrent games played on graphs, in which each player has several qualitative (e.g. reachability or Büchi) objectives, and a preorder on these objectives (for instance the counting order, where the aim is to maximise the number of objectives that are fulfilled).
We study two fundamental problems in that setting: (1) the *value problem*, which aims at deciding the existence of a strategy that ensures a given payoff; (2) the *Nash equilibrium problem*, where we want to decide the existence of a Nash equilibrium (possibly with a condition on the payoffs). We characterise the exact complexities of these problems for several relevant preorders, and several kinds of objectives.

1 Introduction

Games (and especially games played on graphs) have been intensively used in computer science as a powerful way of modelling interactions between several computerised systems [19,9]. Until recently, more focus had been put on the study of purely antagonistic games (a.k.a. zero-sum games), useful for modelling systems evolving in a (hostile) environment.

In the last ten years, non-zero-sum games have entered the picture: they are convenient for modelling complex infrastructures where each individual system tries to fulfil its objectives, while still being subject to uncontrollable actions of the surrounding systems. As an example, consider a wireless network in which several devices try to send data: each device can modulate its transmit power, in order to maximise its bandwidth and reduce energy consumption as much as possible. In that setting, focusing only on optimal strategies for one single agent may be too narrow, and several other solution concepts have been defined and studied in the literature, of which Nash equilibrium [15] is the most prominent. A Nash equilibrium is a strategy profile where no player can improve her payoff by unilaterally changing her strategy, resulting in a configuration of the network that is satisfactory to everyone. Notice that Nash equilibria need not exist or be unique, and are not necessarily optimal: Nash equilibria where all players lose may coexist with more interesting Nash equilibria.

Our contributions. In this paper, we extend our previous study of pure-strategy Nash equilibria in concurrent games with qualitative objectives [2,3] to a (semi-) quantitative setting: we assume that each player is given a set S of qualitative objectives (reachability, for instance), together with a preorder on 2^S. This preorder defines a preference relation (or payoff), and the aim of a player is to maximise her payoff. For instance the counting preorder compares the number of objectives that are fulfilled. As another example, we will consider the lexicographic order, defined in an obvious way once we have ordered the simple objectives. More generally, preorders will be defined by Boolean circuits.

Back to the earlier wireless network example, we can model a simple discretised version of it as follows. From a state each device can increase (action $+$) or keep unchanged (action $=$) its power: the arena of the game is represented for two devices and two levels of power on Fig. 1 (we omit self-loops labelled with $\langle =, = \rangle$). This yields a new bandwidth allocation (which depends on the degradation due to the other devices) and a new energy consumption. The satisfaction of each device is measured as a compromise between energy consumption and bandwidth allocated, and it is given by a quantitative payoff function. This can be transformed into Büchi conditions and a preorder on them.

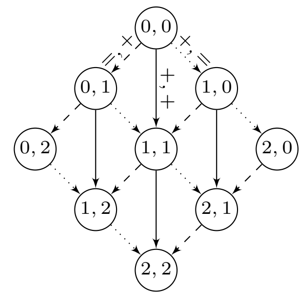


Fig. 1. A simple game-model for the wireless network (states are power levels)

We characterise the exact complexity of deciding the existence of a Nash equilibrium, for reachability and Büchi objectives, under arbitrary preorders. Our techniques also provide us with solutions to the value problem, which corresponds to the purely antagonistic setting described above. We prove for instance that both problems are PSPACE-complete for reachability objectives together with a lexicographic order on these objectives (or for the more general class of preorders defined by Boolean circuits). On the other hand, we show that for sets of Büchi objectives (assumed to be indexed) ordered by the maximum index they contain, both problems are solvable in PTIME.

Related work. Even though works on concurrent games go back to the fifties [18], the complexity of computing Nash equilibria in games played on graphs has only recently been addressed [6,20]. Most of the works so far have focused on turn-based games and on qualitative objectives, but have also considered the more general setting of stochastic games or randomised strategies. Our restriction to pure strategies is justified by the undecidability of computing (randomised) Nash equilibria in concurrent games with quantitative objectives [21]. Though the setting is turn-based, the most relevant related work is [17], where a first step towards quantitative objectives is made: they consider generalised Muller games (with a preference order on the set of states that are visited infinitely often), they show that pure Nash equilibria always exist, and they give a doubly-exponential algorithm for computing a Nash equilibrium. Generalised Muller conditions can be expressed using Büchi conditions and Boolean circuits (which in the worst-case can be exponential-size): from our results we derive an EXPSPACE upper bound for deciding the existence of (and computing) a Nash equilibrium with a constraint on its payoff.

2 Preliminaries

2.1 Concurrent Games.

Definition 1 ([1]). *A (finite)* concurrent game *is a tuple* $\mathcal{G} = \langle \mathrm{States}, \mathrm{Agt}, \mathrm{Act}, \mathrm{Mov}, \mathrm{Tab}\rangle$*, where* States *is a (finite) set of states,* Agt *is a finite set of players,* Act *is a finite set of actions, and*

- $\mathrm{Mov}\colon \mathrm{States}\times\mathrm{Agt} \to 2^{\mathrm{Act}}\setminus\{\varnothing\}$ *is a mapping indicating the actions available to a given player in a given state;*
- $\mathrm{Tab}\colon \mathrm{States}\times\mathrm{Act}^{\mathrm{Agt}} \to \mathrm{States}$ *associates with a given state and a given move*[1] *of the players the resulting state.*

Fig. 2 displays an example of a concurrent game. Transitions are labelled with the moves that trigger them. We say that a *move* $m_{\mathrm{Agt}} = \langle m_A\rangle_{A\in\mathrm{Agt}} \in \mathrm{Act}^{\mathrm{Agt}}$ is *legal* at s if $m_A \in \mathrm{Mov}(s, A)$ for all $A \in \mathrm{Agt}$. A game is *turn-based* if for each state the set of allowed moves is a singleton for all but at most one player.

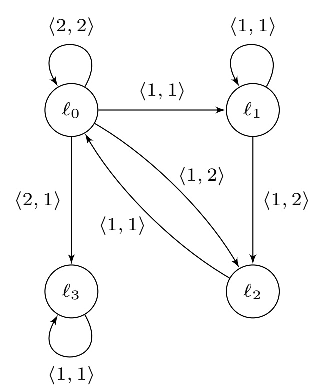


Fig. 2. Example of a two-player concurrent game $\mathcal{A}$

In a concurrent game $\mathcal{G}$, whenever we arrive at a state s, the players simultaneously select an available action, which results in a legal move m_{Agt}; the next state of the game is then $\mathrm{Tab}(s, m_{\mathrm{Agt}})$. The same process repeats *ad infinitum* to form an infinite sequence of states.

A *path* π in $\mathcal{G}$ is a sequence $(s_i)_{0\le i<n}$ (where $n \in \mathbb{N} + \cup\{\infty\}$) of states. The *length* of π, denoted by $|\pi|$, is $n-1$. The set of finite paths (also called *histories*) of $\mathcal{G}$ is denoted by $\mathrm{Hist}_{\mathcal{G}}$, the set of infinite paths (also called *plays*) of $\mathcal{G}$ is denoted by $\mathrm{Play}_{\mathcal{G}}$, and $\mathrm{Path}_{\mathcal{G}} = \mathrm{Hist}_{\mathcal{G}} \cup \mathrm{Play}_{\mathcal{G}}$ is the set of paths of $\mathcal{G}$. Given a path $\pi = (s_i)_{0\le i<n}$ and an integer $j < n$, the *j-th prefix* (resp. *j-th suffix*, *j-th state*) of π, denoted by $\pi_{\le j}$ (resp. $\pi_{\ge j}$, $\pi_{=j}$), is the finite path $(s_i)_{0\le i<j+1}$ (resp. $(s_i)_{j\le i<n}$, state s_j). If $\pi = (s_i)_{0\le i<n}$ is a history, we write $\mathrm{last}(\pi) = s_{|\pi|}$. In the sequel, we write $\mathrm{Hist}_{\mathcal{G}}(s)$, $\mathrm{Play}_{\mathcal{G}}(s)$ and $\mathrm{Path}_{\mathcal{G}}(s)$ for the respective subsets of paths starting in state s. If π is a play, $\mathrm{Occ}(\pi) = \{s \mid \exists j.\ \pi_{=j} = s\}$ is the sets of states that appears at least once along π and $\mathrm{Inf}(\pi) = \{s \mid \forall i.\ \exists j \ge i.\ \pi_{=j} = s\}$ is the sets of states that appears infinitely often along π.

Definition 2. *Let* $\mathcal{G}$ *be a concurrent game, and* $A \in \mathrm{Agt}$*. A* strategy *for* A *is a mapping* $\sigma_A\colon \mathrm{Hist}_{\mathcal{G}} \to \mathrm{Act}$ *such that* $\sigma_A(\pi) \in \mathrm{Mov}(\mathrm{last}(\pi), A)$ *for all* $\pi \in \mathrm{Hist}_{\mathcal{G}}$*.*

[1] A *move* is an element of $\mathrm{Act}^{\mathrm{Agt}}$.

A strategy σ_P for a coalition $P \subseteq \mathrm{Agt}$ is a tuple of strategies, one for each player in P. We write $\sigma_P = (\sigma_A)_{A\in P}$ for such a strategy. A strategy profile *is a strategy for* Agt. *We write* $\mathrm{Strat}^P_{\mathcal{G}}$ *for the set of strategies of coalition P, and* $\mathrm{Prof}_{\mathcal{G}} = \mathrm{Strat}^{\mathrm{Agt}}_{\mathcal{G}}$.

Note that we only consider *pure* (i.e., non-randomised) strategies. Notice also that strategies are based on the sequences of visited states, and not on the sequences of actions played by the players. This is realistic when considering multi-agent systems, where only the global effect of the actions of the agents may be observable. When computing Nash equilibria, this restriction makes it more difficult to detect which players have deviated from their strategies.

Let $\mathcal{G}$ be a game, P a coalition, and σ_P a strategy for P. A path π is *compatible* with the strategy σ_P if, for all $k < |\pi|$, there exists a move m_{Agt} such that

1. m_{Agt} is legal at $\pi_{=k}$,
2. $m_A = \sigma_A(\pi_{\leq k})$ for all $A \in P$, and
3. $\mathrm{Tab}(\pi_{=k}, m_{\mathrm{Agt}}) = \pi_{=k+1}$.

We write $\mathrm{Out}_{\mathcal{G}}(\sigma_P)$ for the set of paths (called the *outcomes*) in $\mathcal{G}$ which are compatible with strategy σ_P of P. We write $\mathrm{Out}^{\mathrm{f}}_{\mathcal{G}}$ (resp. $\mathrm{Out}^{\infty}_{\mathcal{G}}$) for the finite (resp. infinite) outcomes, and $\mathrm{Out}_{\mathcal{G}}(s, \sigma_P)$, $\mathrm{Out}^{\mathrm{f}}_{\mathcal{G}}(s, \sigma_P)$ and $\mathrm{Out}^{\infty}_{\mathcal{G}}(s, \sigma_P)$ for the respective sets of outcomes of σ_P with initial state s. Notice that any strategy profile has a single infinite outcome from a given state.

2.2 Winning objectives

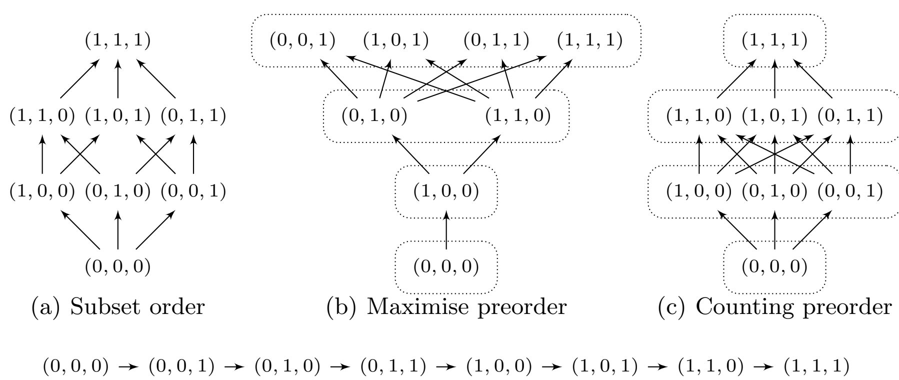
Objectives and preference relations. An *objective* (or *winning condition*) is an arbitrary set of plays. With a set T of states, we associate an objective $\Omega(T)$ in three different ways:

$$\Omega(T) = \{\rho \in \mathrm{Play}_{\mathcal{G}} \mid \mathrm{Occ}(\rho) \cap T \neq \varnothing\} \qquad \text{(Reachability)}$$
$$\Omega(T) = \{\rho \in \mathrm{Play}_{\mathcal{G}} \mid \mathrm{Occ}(\rho) \cap T = \varnothing\} \qquad \text{(Safety)}$$
$$\Omega(T) = \{\rho \in \mathrm{Play}_{\mathcal{G}} \mid \mathrm{Inf}(\rho) \cap T \neq \varnothing\} \qquad \text{(Büchi)}$$

In our setting, each player A is assigned a tuple of such objectives $(\Omega_i)_{1\leq i\leq n}$, together with a preorder $\lesssim$ on $\{0,1\}^n$. Any play ρ then defines a *payoff vector* $\mathbb{1}_{\{i|\rho\in\Omega_i\}} \in \{0,1\}^n$ for player A ($\mathbb{1}_S$ is the vector v such that $v_i = 1 \Leftrightarrow i \in S$; we write $\mathbb{1}$ for $\mathbb{1}_{[1,n]}$, and $\mathbb{0}$ for $\mathbb{1}_{\varnothing}$). The preorder $\lesssim$ then defines another preorder $\precsim$ on the set of plays of $\mathcal{G}$, called the *preference relation* of A, by ordering the plays according to their payoffs: $\rho' \precsim \rho$ if and only if $\mathbb{1}_{\{i|\rho'\in\Omega_i\}} \lesssim \mathbb{1}_{\{i|\rho\in\Omega_i\}}$. Intuitively, the aim of each player is to ensure a play she prefers most.

Examples of preorders. We now describe some preorders on $\{0,1\}^n$ that we consider in the sequel (Fig. 3(a)–3(d) display four of these preorders for $n = 3$). For the purpose of these definitions, we assume that $\max \varnothing = -\infty$.

– *Conjunction*: $v \lesssim w$ iff either $v_i = 0$ for some $0 \leq i \leq n$, or $w_i = 1$ for all $0 \leq i \leq n$. This corresponds to the case where a player wants to achieve all her objectives.
– *Disjunction*: $v \lesssim w$ iff either $v_i = 0$ for all $0 \leq i \leq n$, or $w_i = 1$ for some $0 \leq i \leq n$. The aim here is to satisfy at least one objective.
– *Counting*: $v \lesssim w$ iff $|\{i \mid v_i = 1\}| \leq |\{i \mid w_i = 1\}|$. The aim is to maximise the number of conditions that are satisfied;
– *Subset*: $v \lesssim w$ iff $\{i \mid v_i = 1\} \subseteq \{i \mid w_i = 1\}$: in this setting, a player will always struggle to satisfy a larger (for inclusion) set of objectives.
– *Maximise*: $v \lesssim w$ iff $\max\{i \mid v_i = 1\} \leq \max\{i \mid w_i = 1\}$. The aim is to maximise the highest index of the objectives that are satisfied.
– *Lexicographic*: $v \lesssim w$ iff either $v = w$, or there is an index i such that $v_i = 0$, $w_i = 1$ and $v_j = w_j$ for all $0 \leq j < i$.
– *Parity*: $v \lesssim w$ iff either $\max\{i \mid w_i = 1\}$ is even, or $\max\{i \mid v_i = 1\}$ is odd (or $-\infty$). Combined with reachability objectives, this corresponds to a *weak parity condition*; parity games as they are classically defined correspond to parity preorder with Büchi objectives.
– *Boolean Circuit*: given a Boolean circuit, with input from $\{0,1\}^{2n}$, $v \lesssim w$ if and only if the circuit evaluates 1 on input $v_1 \ldots v_n w_1 \ldots w_n$.
– *Monotonic Boolean Circuit*[2]: same as above, with the restriction that the input gates corresponding to v are negated, and no other negation appear in the circuit.



(a) Subset order (b) Maximise preorder (c) Counting preorder

(d) Lexicographic order

Fig. 3. Examples of preorders (for $n = 3$): dotted boxes represent equivalence classes for the relation $\sim$, defined as $a \sim b \Leftrightarrow a \lesssim b \wedge b \lesssim a$; arrows represent the preorder relation $\lesssim$, forgetting about $\sim$-equivalent elements

[2] Notice that this definition is not exactly the definition of a monotonic circuit, but the idea is the same. Here we negate the input gates of v, so that the value of the circuit increases as u increases and as v decreases.

In terms of expressiveness, any preorder over $\{0,1\}^n$ can be given as a Boolean circuit: for each pair (v,w) with $v \lesssim w$, it is possible to construct a circuit whose output is 1 if and only if the input is $v_1 \dots v_n w_1 \dots w_n$; taking the disjunction of all these circuits we obtain a Boolean circuit defining the preorder. Its size can be bounded by $2^{2n+3}n$, which is exponential in general, but all of our examples can be specified with a circuit of polynomial size.

A preorder $\lesssim$ is *monotonic* if it is compatible with the subset ordering, i.e. if $\{i \mid v_i = 1\} \subseteq \{i \mid w_i = 1\}$ implies $v \lesssim w$. Hence, a preorder is monotonic if fulfilling more objectives never results in a lower payoff. All our non-parametrised examples of preorders (except for the parity preorder), as well as monotonic Boolean circuit preorders, are monotonic. Moreover, any monotonic preorder can be expressed as a monotonic Boolean circuit: for a pair (v,w) with $v \lesssim w$, we can build a circuit whose output is 1 if, and only if, the input is $v_1 \dots v_n w_1 \dots w_n$. We can require this circuit to have negation at the leafs. Now, if the input w_j appears negated, and if $w_j = 0$, then by monotonicity, also the input $(v, \tilde{w})$ is accepted, with $\tilde{w}_i = w_i$ when $i \neq j$ and $\tilde{w}_j = 1$. Hence the negated input gate can be replaced with `true`. Similarly for positive occurrences of any v_j. Hence any monotonic preorder can be written as a monotonic Boolean circuit.

2.3 Nash Equilibria

Given a move m_{Agt} and an action m' for some player B, we write $m_{\mathrm{Agt}}[B \mapsto m']$ for the move n_{Agt} with $n_A = m_A$ when $A \neq B$ and $n_B = m'$. This is extended to strategies in the natural way.

Definition 3. *Let $\mathcal{G}$ be a concurrent game with preference relations $(\precsim_A)_{A\in\mathrm{Agt}}$, and let s be a state of $\mathcal{G}$. A* Nash equilibrium *of $\mathcal{G}$ from s is a strategy profile $\sigma_{\mathrm{Agt}} \in \mathrm{Prof}_{\mathcal{G}}$ such that $\mathrm{Out}(s, \sigma_{\mathrm{Agt}}[B \mapsto \sigma']) \precsim_B \mathrm{Out}(s, \sigma_{\mathrm{Agt}})$ for all players $B \in \mathrm{Agt}$ and all strategies $\sigma' \in \mathrm{Strat}^B$.*

Hence, Nash equilibria are strategy profiles where no player has an incentive to unilaterally deviate from her strategy.

Remark 4. Another possible way of defining Nash equilibrium would be to require that either $\mathrm{Out}(s, \sigma_{\mathrm{Agt}}[B \mapsto \sigma']) \precsim_B \mathrm{Out}(s, \sigma_{\mathrm{Agt}})$, or $\mathrm{Out}(s, \sigma_{\mathrm{Agt}}) \not\precsim_B \mathrm{Out}(s, \sigma_{\mathrm{Agt}}[B \mapsto \sigma'])$. This definition is not equivalent to the one we adopted if the preorder is not total, but both can be meaningful. Notice that with our Definition 3, any Nash equilibrium σ_{Agt} for the subset preorder is also a Nash equilibrium for any monotonic preorder.

2.4 Decision problems

Given a game $\mathcal{G} = \langle \mathrm{States}, \mathrm{Agt}, \mathrm{Act}, \mathrm{Mov}, \mathrm{Tab} \rangle$, a type of objective (Reachability, Safety or Büchi), a preorder on $\{0,1\}^n$, a list of targets for each player $(T_i^A)_{A\in\mathrm{Agt}, i\in\{1,\dots,n_A\}}$, and a state $s \in \mathrm{States}$, we consider the following problems:

– *Value*: Given a player A and a payoff vector v, can player A *ensure* payoff v, i.e., is there a strategy σ_A for player A such that any any outcome in $\mathcal{G}$ from s of σ_A has payoff at least as good (for A) as v?
– *Existence*: Does there exist a Nash equilibrium in $\mathcal{G}$ from s?
– *Constrained existence*: Given two vectors u^A and w^A for each player A, does there exist a Nash equilibrium in $\mathcal{G}$ from s with some payoff $(v^A)_{A\in\mathrm{Agt}}$ satisfying the constraint, *i.e.*, $u^A \lesssim_A v^A \lesssim_A w^A$ for all $A \in \mathrm{Agt}$?

3 Preliminary Lemma

We first characterise outcomes of Nash equilibria as ultimately periodic runs.

Lemma 5. *Assume that every player has a preference relation which only depends on the set of states that are visited, and the set of states that are visited infinitely often, i.e. if* $\mathrm{Inf}(\rho) = \mathrm{Inf}(\rho')$ *and* $\mathrm{Occ}(\rho) = \mathrm{Occ}(\rho')$ *then* $\rho \sim_A \rho'$ *for every player* $A \in \mathrm{Agt}$. *If there is a Nash equilibrium with payoff* v, *then there is a Nash equilibrium with payoff* v *for which the outcome is of the form* $\pi \cdot \tau^\omega$, *where* $|\pi|, |\tau| < |\mathrm{States}|^2$.

In particular, this lemma applies to any preorder over reachability, safety, or Büchi objectives.

Proof. Let σ_{Agt} be a Nash equilibrium, and ρ be its outcome. We will define a new strategy profile σ'_{Agt}, whose outcome is ultimately periodic, and will show that this is a Nash equilibrium as well.

We construct a history $\pi_{\leq n} = \pi_0\pi_1 \dots \pi_n$ which visits precisely those states that are visited by ρ, and which is not too long.

We first set its initial state to be the same: $\pi_0 = \rho_0$. Then we assume we have constructed $\pi_{\leq k} = \pi_0 \dots \pi_k$ which visits exactly the same states as $\rho_{\leq k'}$ for some k'. If all the states of ρ have been visited in $\pi_{\leq k}$ then the construction is finished. Otherwise there is an index i such that ρ_i does not appear in $\pi_{\leq k}$. We therefore define our next target as the smallest such i: we define $t(\pi_{\leq k}) = \min\{i \mid \forall j \leq k.\ \pi_j \neq \rho_i\}$. We then look at the occurrence of the state π_k where we currently are, that is the closest to the target in ρ: we define $c(\pi_{\leq k}) = \max\{i < t(\pi_{\leq k}) \mid \pi_k = \rho_i\}$. Then we emulate what happens at that position by choosing $\pi_{i+1} = \rho_{c(\pi_{\leq i})+1}$. Then π_{i+1} is either the target, or a state that was already seen before in $\pi_{\leq k}$, in which case the constructed $\pi_{\leq k+1}$ visits exactly the same states as $\rho_{\leq c(\pi_{\leq i})+1}$.

At each step, either the number of remaining targets strictly decreases, or the number of remaining targets is constant but the distance to the next target strictly decreases. Therefore the construction terminates. Moreover, notice that between two targets we do not visit twice the same state, and we visit only states that have already been visited, plus the target. As the number of targets is bounded by $|\mathrm{States}|$, we can show that the π constructed so far has a length bounded by $\frac{|\mathrm{States}|\cdot(|\mathrm{States}|-1)}{2} + 1$.

We will now construct $\pi_{\leq m} = \pi_0\pi_1 \dots \pi_m$ which visits precisely those states which are seen infinitely often along ρ, and which is not too long. Let l be the least index s.t. after this point, the states visited by ρ are visited infinitely often: $l = \min\{i \mid \forall j \geq i.\ \rho_j \in \mathrm{Inf}(\rho)\}$. The run $\rho_{\geq l}$ is such that its set of visited states and its set of states visited infinitely often coincide. We therefore define $\pi_{\leq m}$ in the same way we would have defined $u_{\leq m}$ for the run $\rho_{\geq l}$.

We now need to glue together π and τ, and to ensure τ can be glued to itself, so that $\pi \cdot \tau^\omega$ is a real run. We therefore need to link the last state of $\pi_{\leq n}$ with the first state of $\tau_{\leq m}$ (and similarly the last state of $\tau_{\leq m}$ with its first state). It is easy to do: we fix the target of $\pi_{\leq n}$ and $\tau_{\leq m}$ to be π_0. For $k \geq n$ such that none of $\pi_n, \ldots, \pi_k$ is equal to τ_0, we set $t(\pi_{\leq k}) = \min\{i \geq l \mid \rho_i = \tau_0\}$ and as before, we set $c(\pi_{\leq k}) = \max\{i < t(\pi_{\leq k}) \mid \pi_k = \rho_i\}$ and $\pi_{i+1} = \rho_{c(\pi_{\leq i})+1}$. The length we add to the history is again bounded by $|\mathrm{States}|-1$. So the total length of π is bounded by $\frac{(|\mathrm{States}|-1)\cdot(|\mathrm{States}|+2)}{2}+1$ which is smaller than $|\mathrm{States}|^2$. The construction for τ is similar.

We will now define our new strategy profile, that will follow the run $\pi \cdot \tau^\omega$. Given a history h:

- if h followed the expected path, i.e. $h = \pi_{\leq k}$ for some k, or $h = \pi \cdot \tau^{k'} \cdot \tau_{\leq k}$ for some k and k', we mimic the strategy at $c(h)$: $\sigma'_{\mathrm{Agt}}(h) = \sigma_{\mathrm{Agt}}(\rho_{c(\pi_{\leq k})})$ in the first case, and $\sigma'_{\mathrm{Agt}}(h) = \sigma_{\mathrm{Agt}}(\rho_{c'(\tau_{\leq k})})$ in the second one (where c' is the c function associated with τ);
- otherwise we take the longest prefix $h_{\leq k}$ that is of the form described above and define $\sigma'_{\mathrm{Agt}}(h) = \sigma_{\mathrm{Agt}}(\rho_{c(h_{\leq k})} \cdot h_{\geq k+1})$.

We have that $\mathrm{Tab}(\pi_k, \sigma'_{\mathrm{Agt}}(\pi_{\leq k}) = \mathrm{Tab}(\rho_{c(\rho_{\leq k})}, \sigma_{\mathrm{Agt}}) = \pi_{k+1}$ (or τ_0 if $k = |\pi|$) and $\mathrm{Tab}(\pi_k \cdot \tau^{k'} \cdot \tau_{\leq k}, \sigma'_{\mathrm{Agt}}(\pi_{\leq k}) = \mathrm{Tab}(\rho_{c(\pi_n \cdot \tau_k)}, \sigma_{\mathrm{Agt}}) = \tau_{k+1}$ (or τ_0 if $k = |\tau|$), therefore $\pi \cdot \tau^\omega$ is indeed the outcome of σ'_{Agt}.

We now show that σ'_{Agt} is a Nash equilibrium. Assume that one of the players changes her strategy: either the resulting outcome does not deviate from $\pi \cdot \tau^\omega$, in which case the payoff of that player is not improved; or it deviates at some point, and from that point on, σ'_{Agt} follows the same strategies as in σ_{Agt}. Therefore σ'_{Agt} is a Nash equilibrium. □

4 Reachability objectives

4.1 Single objective

Multiplayer games with one reachability objective per player have been studied in [2], where the existence and constrained existence problems are shown NP-complete. The following proposition summarises the complexity results.

Proposition 6 ([2]). *For a single reachability objective per player (with preorder $0 \lesssim_A 1$ for every player A), the existence and constrained existence problems are* NP*-complete in general, but* P*-complete for a bounded number of players.*

The NP algorithms proceed by guessing an ultimately-periodic path (as described in Lemma 5) and then check that it is an outcome of a Nash equilibrium. This last step is done in polynomial time thanks to [2, Prop. 13], which we rephrase as follows.

Lemma 7 ([2]). *For a single reachability objective per player with any preorder, given histories π and τ, we can check in polynomial time whether $\pi \cdot \tau^\omega$ is the outcome of a Nash equilibrium.*

We now assume that each player has several reachability objectives. In the general case where the preorders are given as Boolean circuits, we show that the various decision problems are PSPACE-complete, and we even notice that the hardness result holds for several simpler preorders. We then improve this result in a number of cases. The results are summarised in Table 1.

Table 1. Summary of the results for reachability objectives

Preorder	Value problem	(Constrained) existence
Disjunction, Maximise	P-c	NP-c
Parity	P-c [16]	NP-h, in PSPACE
Subset	PSPACE-c	NP-c
Conjunction, Counting, Lexicographic	PSPACE-c	PSPACE-c
(Monotonic) Boolean Circuit	PSPACE-c	PSPACE-c

4.2 General case

In [3] we have seen how safety and reachability winning conditions can be encoded by 1-weak deterministic Büchi automata. We recall below an important property of the games with objectives given as such automata. Thanks to this property we will be able to provide a polynomial space algorithm.

Lemma 8 ([3]). *Consider a game $\mathcal{G}$ with a single objective per player, given by 1-weak deterministic Büchi automata $(\mathcal{A}_A)_{A\in\mathrm{Agt}}$. For all $A \in \mathrm{Agt}$, write ℓ_A for the length of the longest acyclic path in $\mathcal{A}_A$. We furthermore let m be the space needed for deciding (for any A) whether a state of $\mathcal{A}_A$ is final, and whether a state q' of $\mathcal{A}_A$ is the successor of a state q for an input symbol s. Then the constrained existence problem in $\mathcal{G}$ can be decided in space $O\big((|\mathcal{G}|\cdot(\sum_{A\in\mathrm{Agt}} \ell_A) + \sum_{A\in\mathrm{Agt}} \log|\mathcal{A}_A| + m)^2\big)$. Moreover, whether a path of the form $\pi \cdot \tau^\omega$ is the outcome of a Nash equilibrium can also be decided within the same space bound.*

Notice that in the lemma above, there is no need for having an extensive description of the Büchi automata $(\mathcal{A}_A)_{A\in\mathrm{Agt}}$. In the sequel, we apply this lemma for exponential-size Büchi automata for which m can be bounded by $|\mathrm{Agt}|$. Hence the whole algorithm, implemented by building $(\mathcal{A}_A)_{A\in\mathrm{Agt}}$ on-the-fly, runs in polynomial space.

Theorem 9. *For reachability objectives with preorders given by Boolean circuits, the value, existence and constrained existence problems are in* PSPACE*. For* *preorders having $\mathbb{1}$ as a unique maximal element, the value problem is* PSPACE-*complete. If moreover there is an element*[3] $v \in \{0,1\}^n$ *such that* $\mathbb{1} \not\lesssim v' \Leftrightarrow v' \lesssim v$, *then the existence and constrained existence problems are* PSPACE-*complete (even for two-player games).*

Before proving these results, let us first see which of our preorders satisfy the conditions.

Lemma 10. *The conjunction, subset, counting and lexicographic preorders have the unique maximal element* $\mathbb{1}$. *The conjunction, counting and lexicographic preorders have an element* v *such that* $\mathbb{1} \neq v' \Leftrightarrow v' \lesssim v$.

Proof. Clearly enough, $\mathbb{1}$ is a maximal element for these orders, as they are monotonic. Assume that there is another maximal element $m \neq \mathbb{1}$. One easily checks that $m \lesssim \mathbb{1}$ and $\mathbb{1} \not\lesssim m$ for the four preorders under study, which is a contradiction.

Consider $v = (1, \ldots, 1, 0)$, and v' such that $\mathbb{1} \neq v'$. For conjunction, there is i such that $v'_i = 0$, so $v' \lesssim v$. For counting, $|\{i \mid v'_i = 1\}| < n$, so $v' \lesssim v$. For the lexicographic preorder, let i be the smallest index such that $v'_i = 0$, and either $v_i = 1$ and $v_j = v'_j$ for all $j < i$, or $v_j = v'_j$ for all $j \in \{1, \ldots, n\}$. In both cases $v' \lesssim v$. □

As conjunction (for instance) can easily be encoded using a (monotonic) Boolean circuit in polynomial time, the hardness results are also valid if the preorder is given by a (monotonic) Boolean circuit. On the other hand, the disjunction and maximise preorders do not have a unique maximal element, so the hardness result does not carry over to these preorders. In the same way, for the subset preorder, there is no v such that $\mathbb{1} \not\lesssim v' \Leftrightarrow v' \lesssim v$, so the hardness result does not apply. We prove later (in Section 4.3) that in these special cases, the complexity is actually lower.

Proof (of Theorem 9). The rest of Section 4.2 is devoted to the proof of Theorem 9. We begin with a PSPACE algorithm for the constrained existence problem with preorders given as Boolean circuits, and show how it can be used to solve the value problem. We then turn to the PSPACE-hardness proofs in the different cases listed in the Theorem.

Proof of the PSPACE *upper bounds.* We first focus on the constrained existence problem; we fix a game $\mathcal{G}$ with reachability objectives and a preorder for every player, and a constraint on the payoffs. The algorithm enumerates the possible payoff vectors (within the given constraint) and looks for a Nash equilibrium with that payoff. Fix such a payoff tuple $\mathbf{v} = (v^A)_{A \in \mathrm{Agt}}$. We consider a new game $\mathcal{G}(\mathbf{v})$: its structure is the same as that of $\mathcal{G}$, and each player A has a single objective given by a 1-weak deterministic Büchi automaton $\mathcal{A}(v^A)$ (defined below). The

[3] To be fully formal, the preorder $\lesssim$ is in fact a family $(\lesssim_n)_{n \in \mathbb{N}}$ (where $\lesssim_n$ compares two vectors of size n), and this condition should be stated as "*if, for all n, there is an element $v_n \in \{0,1\}^n$ such that for all $v' \in \{0,1\}^n$, it holds $\mathbb{1} \not\lesssim_n v' \Leftrightarrow v' \lesssim_n v$*".

new game satisfies the following property: there is a Nash equilibrium in $\mathcal{G}$ with payoff $\mathbf{v}$ if, and only if, there is a Nash equilibrium in $\mathcal{G}(\mathbf{v})$ whose outcome has payoff $(0,\ldots,0)$ in $\mathcal{G}(\mathbf{v})$, and payoff $\mathbf{v}$ in $\mathcal{G}$ (notice that runs in $\mathcal{G}(\mathbf{v})$ are also runs in $\mathcal{G}$, as both games only differ in the objectives of the players). Then, trying all the paths of the form described in Lemma 5, and applying Lemma 8, we obtain a polynomial space algorithm to decide the existence of a Nash equilibrium with that payoff, and therefore more generally the constrained existence problem.

We now explain how we construct the automata $\mathcal{A}(v^A)$. All these automata share a common structure, which we denote by $\mathcal{A}$. The set of states of $\mathcal{A}$ is 2^{States}, with, for each set $S \subseteq \text{States}$ and each $s \in \text{States}$, a transition from S to $S \cup \{s\}$ labelled by s. The construction is illustrated by an example in Fig. 4. The initial state is the empty set, and each automaton $\mathcal{A}(v^A)$ is obtained from $\mathcal{A}$ by adding the set of accepting states $F(v^A) = \{S \mid \mathbb{1}_{\{i \mid S \cap T_i^A \neq \varnothing\}} \not\lesssim v^A\}$, where $(T_i^A)_i$ are the targets of player A in game $\mathcal{G}$. While reading a word ρ from the initial state, the current state of $\mathcal{A}(v^A)$ is the set of states that have been seen so far. Hence, the run on ρ will ultimately loop in the state $\text{Occ}(\rho)$. It will end up in $F(v^A)$ iff it is an improvement for player A compared to the payoff v^A. More precisely, if v is the payoff of ρ for player A in $\mathcal{G}$, then $\mathcal{A}(v^A)$ accepts it iff $v \not\lesssim v^A$. With this construction, the announced equivalence is straightforward.

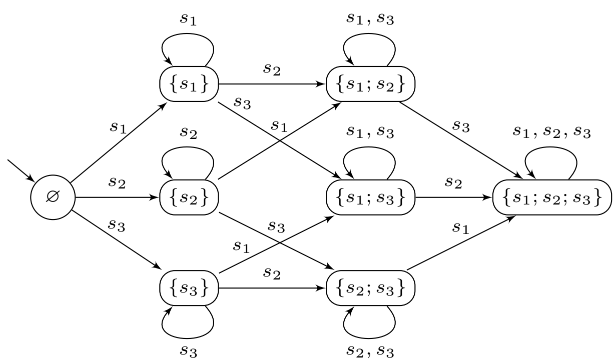


Fig. 4. Automaton $\mathcal{A}$ for set of states $\{s_1, s_2, s_3\}$

We now turn to the proof for the value problem. We fix a game $\mathcal{G}$, a player A and a threshold v. We define a new (polynomial size) two-player game by considering the opponents of A as a single player, and changing the preferences of the players as follows: player A now has no objective, and her opponent (coalition) wins if the payoff is not above v in the original game, i.e. if the run has payoff v' with $v \not\lesssim v'$. Then, there is a Nash equilibrium where the opponent loses iff there is a strategy for A that ensures v in the original game. We can thereby use the algorithm that decides constrained existence.

Proof of PSPACE*-hardness for the value problem.*

Lemma 11. *For reachability objectives and a preorder having $\mathbb{1}$ as a unique maximal element, the value problem for turn-based two-player games is* PSPACE*-hard.*

For the proof, we use the fact that these games are determined, stated as follows:

Lemma 12 ([13]). *Let $\mathcal{G}$ be a turn-based game with reachability objectives and preorder $\lesssim_A$ for player A, $s \in$ States, and $v \in \{0,1\}^n$. Either there exists a strategy for A that ensures payoff v from s, or there exists a strategy σ for the coalition $\mathrm{Agt}\backslash\{A\}$ such that all outcomes of σ from s have payoff v' with $v \not\lesssim_A v'$.*

Proof (of Lemma 11). We reduce from QSAT, the satisfiability problem for quantified Boolean formula, where we assume without loss of generality that the Boolean formula is a conjunction of disjunctive clauses[4].

Let $\phi = Q_1 x_1 \ldots Q_n x_n.\ \phi'$, where $Q_i \in \{\forall, \exists\}$ and $\phi' = C_1 \wedge \cdots \wedge C_m$ with $C_j = \bigwedge_{1\le k\le p} \ell_{j,k}$ and $\ell_{j,k} \in \{x_i, \neg x_i \mid 1 \le i \le n\} \cup \{\top, \bot\}$. We define a turn-based game $\mathcal{G}(\phi)$ in the following way (illustrated in Example 14 below). There is one state for each quantifier, one for each literal, and two additional states $\top$ and $\bot$:

$$\text{States} = \{Q_i \mid 1 \le i \le n\} \cup \{x_j, \neg x_j \mid 1 \le j \le m\} \cup \{\top, \bot\}.$$

The game involves two players, A and B. Both states $\top$ and $\bot$, the existential-quantifier states and the literal states are controlled by A, while the universal-quantifier states belong to player B. The state corresponding to quantifier Q_i has two outgoing transitions, going to x_i and $\neg x_i$ respectively. The literal states only have one transition to the next quantifier state, or to the final state for the last literal state. Finally, states $\top$ and $\bot$ both carry a self-loop (notice that $\bot$ is not reachable, while $\top$ will always be visited).

Player A has one target for each clause: if $C_j = \bigwedge_{1\le k\le p} \ell_{j,k}$ then $T_j^A = \{\ell_{j,k} \mid 1 \le k \le p\}$. The j-th objective Ω_j^A is to reach target T_j^A. The following result is then straightforward:

Lemma 13. *Formula ϕ is valid iff player A has a strategy whose outcomes from state Q_1 all visit each set T_j^A.*

We can directly conclude from this lemma that the value of the game for A is $\mathbb{1}$ (the unique maximal payoff for our preorder) iff the formula ϕ is valid, hence this problem is PSPACE-hard. □

Example 14. As an example of the construction, let us consider the formula

$$\phi = \forall x_1.\ \exists x_2.\ \forall x_3.\ \exists x_4.\ (x_1 \vee \neg x_2 \vee \neg x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge \neg x_4 \qquad (1)$$

The targets for player A are given by $T_1^A = \{x_1; \neg x_2; \neg x_3\}$, $T_2^A = \{x_1; x_2; x_4\}$, and $T_3^A = \{\neg x_4\}$. The structure of the game is represented in Fig. 5. B has a strategy that falsifies one of the clauses whatever A does, which means that the formula is not valid.

[4] With the convention that en empty disjunction is equivalent to $\bot$.

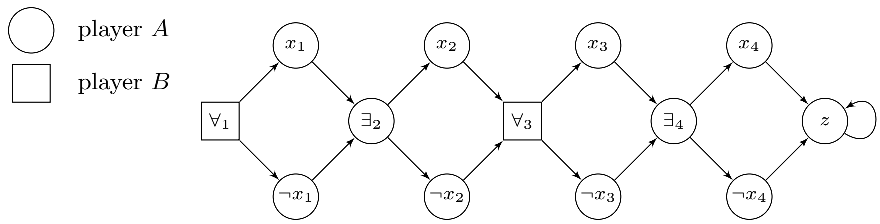


Fig. 5. Reachability game associated with the formula (1)

Proof of PSPACE*-hardness for the (constrained) existence problem.* The previous hardness proof applies in particular to conjunctions of reachability objectives. We use a reduction from this problem to prove that the constrained existence problem is PSPACE-hard, under the conditions specified in the statement of Theorem 9. Let $\mathcal{G}$ be a turn-based game with a conjunction of reachability objectives for player A and v be as defined in Theorem 9. We build a new game $\mathcal{G}'$ from $\mathcal{G}$ by adding an initial state s'_0, and a sink state z. In the initial state s'_0, both players A and B have two moves 0 and 1; if they propose the same move, the game goes to z, otherwise it goes to the initial state of $\mathcal{G}'$. We modify the targets of player A so that, in $\mathcal{G}'$, reaching z exactly gives her payoff v. The new sink state z is the unique target of player B.

We now prove that there is no strategy for A in $\mathcal{G}$ ensuring payoff $\mathbb{1}$ from s_0 if, and only if, there is a Nash equilibrium in $\mathcal{G}'$ with payoff v. First assume there is no strategy for player A in $\mathcal{G}$ from s_0 that ensures payoff $\mathbb{1}$. As turn-based games are determined (Prop. 12), player B has a strategy whose outcomes all have payoff (for A) smaller than or equal to v. We fix this strategy for player B and any strategy for player A in $\mathcal{G}$. In s'_0 we then set for both players a strategy that consists in going to z. It is easy to see that this is a Nash equilibrium, with payoff v for A and payoff 1 for B.

Conversely, assume there is a Nash equilibrium from s'_0. Then it ends up in state z, since otherwise player B would better change her strategy at s'_0. Also, the strategy of player B in that equilibrium secures payoff at most v for player A in $\mathcal{G}$, since otherwise player A would be able to improve her payoff. Hence player A has no strategy to get payoff $\mathbb{1}$ in $\mathcal{G}$. $\square$

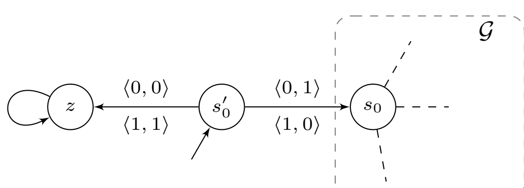


Fig. 6. Reduction from value to existence

4.3 Simple cases

For some preorders, the preference relation can (efficiently) be reduced to a single reachability objective. For instance, a disjunction of several reachability objectives can obviously be reduced to a single reachability objective, by considering the union of the targets. Formally, we say that a preorder $\lesssim$ is *reducible* to a single (reachability) objective if, given any payoff vector v, we can construct in polynomial time a target $\hat{T}^A$ such that for all paths ρ, $v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\}}$ if, and only if, $\mathrm{Occ}(\rho) \cap \hat{T}^A \neq \varnothing$. It means that *securing* payoff v corresponds to ensuring a visit to the new target. Similarly, we say that the preorder is *co-reducible* to a single reachability objective if for any vector v we can construct $\hat{T}^A$ such that $\mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\}} \not\lesssim v$ if, and only if $\mathrm{Occ}(\rho) \cap \hat{T}^A \neq \varnothing$. It means that *improving* on payoff v corresponds to ensuring a visit to the new target.

Lemma 15. *If a total preorder is reducible to a single reachability objective, then it is co-reducible to a single reachability objective.*

Proof. Consider a total preorder which is reducible to a single reachability objective, and let u be a vector. If u is a maximal element, the new target is empty, which satisfies the property for co-reducibility. Otherwise we pick a vector v among the smallest elements that is strictly greater than u. Since the preorder is reducible to a single reachability objective, there is a target $\hat{T}^A$ that is reached whenever the payoff is greater than v. Since the preorder is total and by choice of v, we have $w \not\lesssim u \Leftrightarrow v \lesssim w$. Thus the target $\hat{T}^A$ is visited when u is not greater than the payoff. Hence the preorder is co-reducible to a single reachability objective. □

Proposition 16. *For reachability objectives with a (non-trivial) preorder reducible to a single reachability objective, the value problem is* P*-complete. For a (non-trivial) preorder co-reducible to a single reachability objective, the existence and constrained existence problems are* NP*-complete.*

Proof. Since P-hardness already holds for one reachability objective (see Prop. 6), we only focus on the upper bound.

We begin with the value problem: given a payoff vector u for A, we build the new target $\hat{T}^A$ in polynomial time, and then use a classical algorithm for deciding whether A has a winning strategy (see [8, Chapter 2]). If she does, then she can secure payoff u.

Consider now the constrained existence problem, and assume that the preference relation for each player A are given by target sets $(T_i^A)_{1 \leq i \leq n_A}$. The NP-algorithm consists in guessing the payoff vector $(v_A)_{A \in \mathrm{Agt}}$ and an ultimately periodic play $\rho = \pi \cdot \tau^\omega$ which, for each A, visits T_i^A if, and only if, $v_i^A = 1$. We then reduce the payoff to a new target set $\hat{T}^A$ for each player A.

The run ρ is the outcome of a Nash equilibrium with payoff v_i^A for the original preference relation if, and only if, ρ is the outcome of a Nash equilibrium with payoff 0 with the single reachability objective $\hat{T}^A$ for each $A \in \mathrm{Agt}$. Indeed, in both cases, this is equivalent to the property that no player A can enforce a payoff greater than v^A. Thanks to Prop. 7, this condition can be checked in polynomial time. $\square$

Lemma 17. *Disjunction and maximise preorders are reducible to single reachability objectives. These preorders and the subset order are co-reducible to a single reachability objective.*

Proof. For disjunction, if the payoff v is different from $\mathbb{0}$ then we define $\hat{T}^A$ as the union of all the targets of A: $\hat{T}^A = \bigcup_j T_j^A$. Then, $v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\}}$, iff, there is some i for which $\mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing$, iff $\mathrm{Occ}(\rho) \cap \hat{T}^A \neq \varnothing$. If the payoff v is $\mathbb{0}$ then $\hat{T}^A = \mathrm{States}$. Disjunction being total, it is also co-reducible (from Lemma 15).

We now consider the maximise preoder. Given a payoff v, consider the index $i_0 = \max\{i \mid v_i = 1\}$. We then define $\hat{T}^A$, as the union of the target sets that are above i_0: $\hat{T}^A = \bigcup_{i \geq i_0} T_i^A$. The following four statements are then equivalent:

$$\begin{aligned} v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\}} \quad &\Leftrightarrow \quad i_0 \leq \max\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\} \\ &\Leftrightarrow \quad \exists i \geq i_0.\ \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing \end{aligned}$$

Hence maximise is reducible, and also co-reducible as it is total, to a single reachability objective.

Finally, we prove that the subset preorder is co-reducible to a single reachability objective. Given a payoff v, the new target is the union the targets that are not reached for that payoff: $\hat{T}^A = \bigcup_{\{i \mid v_i = 0\}} T_i^A$. Then the following statements are equivalent:

$$\begin{aligned} \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing\}} \not\lesssim u \quad &\Leftrightarrow \quad \exists i.\ \mathrm{Occ}(\rho) \cap T_i^A \neq \varnothing \text{ and } u_i = 0 \\ &\Leftrightarrow \quad \mathrm{Occ}(\rho) \cap \hat{T}^A \neq \varnothing \end{aligned}$$

$\square$

5 Safety objectives

The results for safety objectives are summarized in Table 2. We prove these results in this section, beginning with a polynomial-space algorithms when the preorder is defined as a Boolean circuit, and characterizing classes of preorders for which PSPACE-hardness holds. We then consider preorders outside those classes, and establish the complexity of the associated problems.

5.1 General case

Theorem 18. *For safety objectives with preorders given as Boolean circuits, the value, existence and constrained existence problems are in* PSPACE. *For preorders having $\mathbb{0}$ as a unique minimal element, the existence and constrained-existence problems are* PSPACE-*complete, even for two players. If additionally there is a vector $v \in \{0,1\}^n$ satisfying the equivalence $v \not\lesssim v' \Leftrightarrow v' = \mathbb{0}$, then the value problem is* PSPACE-*complete.*

Table 2. Summary of the results for safety

Preorder	Value problem	(Constr.) existence
Conjunction	P-c	NP-c
Subset	P-c	PSPACE-c
Disjunction, Parity	PSPACE-c	PSPACE-c
Counting, Maximise, Lexicographic	PSPACE-c	PSPACE-c
(Monotonic) Boolean Circuit	PSPACE-c	PSPACE-c

Proof. The algorithm is readily obtained from the PSPACE algorithm for reachability objectives with a preorder defined by Boolean circuits: indeed, since a safety condition is the negation of reachability condition, by negating the input of the circuit, we obtain the same preference relation, but defined with reachability targets. Hence the PSPACE algorithm.

We now prove hardness of the value problem.

Lemma 19. *For safety objectives and a preorder with $\mathbb{0}$ as a unique minimal element, and where there is a vector $v \in \{0,1\}^n$ such that $v \not\lesssim v' \Leftrightarrow v' = \mathbb{0}$, the value problem in turn-based two-player games is* PSPACE*-hard.*

Proof. We use the fact that for a conjunction of reachability objectives, the problem of verifying that A is winning (with payoff $\mathbb{1}$) is PSPACE-hard for turn-based games (see Lemma 11). Let $\mathcal{G}$ be a turn-based game where player A has a conjunction of reachability objectives. We transform $\mathcal{G}$ into a new game $\mathcal{G}'$ by changing the objectives of the players: the reachability objectives of A are turned into safety objectives for player B.

Lemma 20. *There is a strategy for player A in $\mathcal{G}$ such that any outcome has payoff $\mathbb{1}$ (for the reachability objectives of A) if, and only if, there is no strategy for player B in $\mathcal{G}'$ such that any outcome has payoff at least v (for the safety objectives of B).*

Proof. Assume that player A has a strategy to ensure payoff $\mathbb{1}$ in $\mathcal{G}$: Since turn-based games are determined, player B does not have a strategy to avoid visiting any of the reachability objectives of A. Hence player B cannot ensure a payoff different from $\mathbb{0}$ in $\mathcal{G}$, and she can not secure payoff v in $\mathcal{G}'$.

For the converse direction, we again use the fact that turn-based games are determined. Assuming there is a vector v as described in Lemma 19, there is a strategy for B to secure a payoff at least v in the new game, if and only if, A has no strategy that ensures payoff $\mathbb{1}$ in the original game. This proves the hardness result for the value problem. □

□

Lemma 21. *The maximise, disjunction, counting, and lexicographic preorders all meet the condition of Lemma 19.*

Proof. Consider the vector $v = \mathbb{1}_{\{1\}}$. For the maximise preorder, if $v \not\lesssim v'$ then $\max\{i \mid v'_i = 1\} = 0$, so $v' = \mathbb{0}$. For disjunction, if $v \not\lesssim v'$ then there is no i such that $v'_i = 1$, so $v' = \mathbb{0}$. For the counting preorder, if $v \not\lesssim v'$ then $|\{i \mid v'_i = 1\}| = 0$, so $v' = \mathbb{0}$. Now consider $v = \mathbb{1}_{\{n\}}$. For the lexicographic order, if $v \not\lesssim v'$ then there is j such that $v'_j = 0$, $v_j = 1$, and for all $i < j$, $v'_j = v_j$, so $v' = \mathbb{0}$. Therefore all these preorders meet the condition of Lemma 19. □

Lemma 22. *For safety objectives and a preorder with unique minimal element* $\mathbb{0}$*, the existence problem for concurrent two-player games is* PSPACE*-hard.*

Proof. We use a reduction from the same problem as in the proof of Lemma 19. Let $\mathcal{G}$ be a two-player turn-based game. We build a new game $\mathcal{G}'$ such that there is a correspondence between the value problem from state s_0 in $\mathcal{G}$ and the existence of a Nash equilibrium in $\mathcal{G}'$ from s'_0. The construction is very similar to the one we used to show the hardness of the existence problem for reachability, only the objectives are different.

We add an initial state s'_0, and a terminal state z. From the initial state, the game proceeds to z when both players propose the same move, and to the initial state s_0 of $\mathcal{G}$ otherwise. The objectives are as follows: player A only has one safety objective, consisting in avoiding all the states of $\mathcal{G}$. The safety objectives of player B are defined from the reachability objectives of A in $\mathcal{G}$ as follows: for each (reachability) objective T_i^A of A in $\mathcal{G}$, B has a (safety) objective $\bar{T}_i^B = T_i^A \cup \{z\}$ in $\mathcal{G}'$.

Assume that there is a strategy for player A in $\mathcal{G}$ which, when played from s_0, ensures a visit to each target set T_i^A. We fix such a strategy σ_A for player A, and consider any strategy for player B in $\mathcal{G}$. We extend both strategies by defining them from s'_0, requiring that both propose the same move from that state. One easily checks that this defines a Nash equilibrium in $\mathcal{G}'$ from s'_0: player A is winning already, so she would not want to change her strategy. Player B has no hope of improving, since going to $\mathcal{G}$ from s'_0 would trigger a visit to each T_i^A, so that her payoff would still be $\mathbb{0}$.

Conversely assume there is a Nash equilibrium. Then it must end up in state z, as otherwise player A would have an incentive to change her strategy at s'_0. Also, it must be the case that player A has a strategy enforcing a visits to each T_i^A from s_0, as otherwise player B would better go to s_0. □

Lemma 23. *The value and existence problem are* PSPACE*-hard for parity preorder.*

Proof. This is a consequence of the hardness results for the disjunction preorder, and of the fact that we can encode any instance of disjunction into parity, by giving as new targets $\bar{T}_1^A = \varnothing$ and $\bar{T}_{2i}^A = T_i^A$. □

□

5.2 Simple cases

We now consider simpler cases. As for reachability, the simple cases are for the preference relations that are *reducible* or *co-reducible* to a single safety objective. The preorder $\lesssim$ is *reducible to a single safety objective*, if given any payoff vector v, we can construct in polynomial time a target $\hat{T}^A$ such that $v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A = \varnothing\}}$ if, and only if $\mathrm{Occ}(\rho) \cap \hat{T}^A = \varnothing$. It is *co-reducible* to a single safety objective if we can construct in polynomial time a target $\hat{T}^A$ such that $\mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A = \varnothing\}} \not\lesssim v$ if, and only if $\mathrm{Occ}(\rho) \cap \hat{T}_A = \varnothing$.

Proposition 24. *For safety objectives, a (non trivial) preorder reducible to a single safety objective, the value problem is* P*-complete. For a (non trivial) preorder co-reducible to a single safety objective, the existence and constrained existence problem are* NP*-complete. In particular value is* P*-complete for conjunction and subset. Existence and constrained existence are* NP*-complete for conjunction.*

Proof. The idea is the same than the one we used for reachability in Prop. 16. For the value problem, we construct the new targets and then use an attractor computation to decide whether A has a winning strategy. For the constrained existence problem, the algorithm guesses a play and the payoff for each player A, reduces the payoff to new targets, and then checks that in the new game there is an equilibrium with payoff 0 for every player. □

Lemma 25. *Conjunction is reducible and co-reducible to a single reachability objective. Subset is reducible to a single reachability objective.*

Proof. We begin with conjunction: for a payoff v different from $\mathbb{1}$, $\hat{T}^A = \varnothing$, as all the runs have a payoff greater or equal to v. For $v = \mathbb{1}$, we define $\hat{T}^A = \bigcup_{\{i \mid u_i = 1\}} T_i^A$. Then the following statements are equivalent:

$$v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A = \varnothing\}} \quad \Leftrightarrow \quad \forall i.\ \mathrm{Occ}(\rho) \cap T_i^A = \varnothing \quad \Leftrightarrow \quad \mathrm{Occ}(\rho) \cap \hat{T}^A = \varnothing$$

This entails reducibility to a single objective. Co-reducibility can be proved by dualizing Lemma 15.

We now turn to the subset preorder. For a payoff v, we define the target $\hat{T}^A = \bigcup_{\{i \mid v_i = 1\}} T_i^A$. Then the following three statements are equivalent:

$$\begin{aligned} v \lesssim \mathbb{1}_{\{i \mid \mathrm{Occ}(\rho) \cap T_i^A = \varnothing\}} \quad &\Leftrightarrow \quad \{i \mid u_i = 1\} \subseteq \{i \mid \mathrm{Occ}(\rho) \cap T_i^A = \varnothing\} \\ &\Leftrightarrow \quad \mathrm{Occ}(\rho) \cap \hat{T}^A = \varnothing \end{aligned}$$

□

6 Büchi objectives

We now turn to Büchi objectives, for which we prove the results listed in Table 3.

6.1 Reduction to a zero-sum game

In this section, we show how, from a multiplayer game $\mathcal{G}$, we can construct a two-player game $\mathcal{H}$, such that there is a correspondence between Nash equilibria in $\mathcal{G}$ and winning strategies in $\mathcal{H}$. This will allow us to reuse algorithmic techniques from zero-sum game to solve our problems.

Table 3. Summary of the results for Büchi objectives

Preorder	Value	Existence	Constr. exist.
Maximise, Disjunction, Subset	P-c	P-c	P-c
Conjunction, Lexicographic	P-c	P-h, in NP	NP-c
Counting	coNP-c	NP-c	NP-c
Monotonic Boolean Circuit	coNP-c	NP-c	NP-c
Parity	UP $\cap$ coUP [11]	coNP-h [3], in $\mathsf{P}_{\parallel}^{\mathsf{NP}}$	$\mathsf{P}_{\parallel}^{\mathsf{NP}}$-c
Boolean Circuit	PSPACE-c	PSPACE-c	PSPACE-c

We begin with introducing a few extra definitions. We say that a strategy profile σ_{Agt} is a *trigger strategy* for payoff $(v^A)_{A\in\mathrm{Agt}}$ from the state s if for any strategy σ'_A of any player $A \in \mathrm{Agt}$, the outcome ρ of $\sigma_{\mathrm{Agt}}[A \mapsto \sigma'_A]$ from s satisfies $\mathbb{1}_{\{i \mid \rho\in\Omega_i^A\}} \lesssim v^A$.

Remark 26. A Nash equilibrium is a trigger strategy for the payoff of its outcome. Reciprocally, if σ_{Agt} has payoff $(v^A)_{A\in\mathrm{Agt}}$ and it is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$, then it is a Nash equilibrium.

Given two states s and s', and a move m_{Agt}, the set of *suspect players* [2] for (s, s') and m_{Agt}, denoted with $\mathrm{Susp}((s, s'), m_{\mathrm{Agt}})$, is the set

$$\{A \in \mathrm{Agt} \mid \exists\, m' \in \mathrm{Mov}(s, A).\ \mathrm{Tab}(s, m_{\mathrm{Agt}}[A \mapsto m']) = s'\}.$$

Intuitively, player $A \in \mathrm{Agt}$ is a suspect for transition (s, s') and move m_{Agt} if she can unilaterally change her action to activate the transition to s'. Obviously, if $\mathrm{Tab}(s, m_{\mathrm{Agt}}) = s'$, then $\mathrm{Susp}((s, s'), m_{\mathrm{Agt}}) = \mathrm{Agt}$. Now, given a play ρ and a strategy profile σ_{Agt}, the set of suspect players for ρ and σ_{Agt}, written $\mathrm{Susp}(\rho, \sigma_{\mathrm{Agt}})$ is the set of players that are suspect along each transition of ρ, i.e.,

$$\Big\{A \in \mathrm{Agt} \;\Big|\; \forall i \leq |\rho|\,.\ A \in \mathrm{Susp}\big((\rho_{=i}, \rho_{=i+1}), \sigma_{\mathrm{Agt}}(\rho_{\leq i})\big)\Big\}.$$

Then player A is in $\mathrm{Susp}(\rho, \sigma_{\mathrm{Agt}})$ if, and only if, there is a strategy σ'_A such that $\mathrm{Out}(s, \sigma_{\mathrm{Agt}}[A \mapsto \sigma'_A]) = \rho$.

With a game $\mathcal{G}$ and a payoff $(v^A)_{A\in\mathrm{Agt}}$, we associate a two-player turn-based game $\mathcal{H}(\mathcal{G}, (v^A)_{A\in\mathrm{Agt}})$ (which we write $\mathcal{H}$ when $\mathcal{G}$ and $(v^A)_{A\in\mathrm{Agt}}$ are clear from the context). The set V_1 of vertices of $\mathcal{H}$ controlled by Player A_1 is (a subset of) $\mathrm{States} \times 2^{\mathrm{Agt}}$, and the set V_2 of states controlled by Player A_2 is (a subset of) $\mathrm{States} \times 2^{\mathrm{Agt}} \times \mathrm{Act}^{\mathrm{Agt}}$. The game is played in the following way: from a configuration (s, P) in V_1, player A_1 chooses a legal move m_{Agt} from s; the next state is (s, P, m_{Agt}); then player A_2 choose some state s' in States, and the new configuration is $(s', P \cap \mathrm{Susp}((s, s'), m_{\mathrm{Agt}}))$. In particular, when the state s' chosen by player A_2 is such that $s' = \mathrm{Tab}(s, m_{\mathrm{Agt}})$ (we say that A_2 *obeys* A_1), then the new configuration is (s', P).

We define projections π_1 and π_2 from V_1 on States and 2^{Agt}, resp., by $\pi_1(s,P) = s$ and $\pi_2(s,P) = P$. We extend these projections to plays in a natural way (but only using player A_1 states to avoid stutter), letting $\pi_1((s_0,P_0)\cdot(s_0,P_0,m_0)\cdot(s_1,P_1)\cdots) = s_0\cdot s_1\cdots$. For any run ρ, $\pi_2(\rho)$ (seen as a sequence of sets of players) is decreasing, therefore its limit $L(\rho)$ is well defined. An outcome ρ is *winning for player* A_1, if for all $A \in L(\rho)$, $\mathbb{1}_{\{i \mid \pi_1(\rho)\in\Omega_i^A\}} \lesssim v^A$. In general, since Ω_i^A define Büchi conditions, the winning condition for A_1 can be represented using a (possibly exponential-size[5]) Muller condition. The *winning region* $W(\mathcal{G},(v^A)_{A\in\mathrm{Agt}})$ (later simply denoted by W when $\mathcal{G}$ and $(V^A)_{A\in\mathrm{Agt}}$ are clear form the context) is the set of configurations of $\mathcal{H}(\mathcal{G},(v^A)_{A\in\mathrm{Agt}})$ from which A_1 has a winning strategy. Intuitively player A_1 tries to have the players play a Nash equilibrium, and player A_2 tries to disprove that it is a Nash equilibrium, by finding a possible deviation that improves the payoff of one of the players.

At first sight, the number of states in $\mathcal{H}$ is exponential (in the number of players). However, there are two cases for which we easily see that this is not the case:

- if there is a state in which all the players have several possible moves, then the transition table (which is part of the input [12]) is also exponential in the number of players.
- if the game is turn-based, then the transition table is "small", but also there is always at most one suspect player (unless all of them are suspects), so that the number of reachable states in $\mathcal{H}$ is small.

We now prove that this can be generalized:

Lemma 27. *The number of reachable configurations from* States $\times$ $\{\mathrm{Agt}\}$ *in* $\mathcal{H}$ *is polynomial in the size of* $\mathcal{G}$*.*

Proof. The game $\mathcal{H}$ contains the state (s,Agt) and the states $(s,\mathrm{Agt},m_{\mathrm{Agt}})$, where m_{Agt} is a legal move from s; the number of these states is bounded by $|\mathrm{States}| + |\mathrm{Tab}|$. The successors of those states that are not of the same form are the states $(t,\mathrm{Susp}((s,t),m_{\mathrm{Agt}}))$ with $t \neq \mathrm{Tab}(s,m_{\mathrm{Agt}})$. If some player B is a suspect for transition (s,t) and move vector m_{Agt}, then besides m_B, she must have at least a second action m', for which $\mathrm{Tab}(s,m_{\mathrm{Agt}}[B\mapsto m']) = t$. Thus the transition table from state s has size at least $2^{|\mathrm{Susp}((s,t),m_{\mathrm{Agt}})|}$. The successors of $(t,\mathrm{Susp}((s,t),m_{\mathrm{Agt}}))$ are of the form (t',P) or (t',P,m_{Agt}) where P is a subset of $\mathrm{Susp}((s,t),m_{\mathrm{Agt}})$; there can be no more than $(|\mathrm{States}|+|\mathrm{Tab}|)\cdot 2^{|\mathrm{Susp}((s,t),m_{\mathrm{Agt}})|}$ of them, which is bounded by $(|\mathrm{States}|+|\mathrm{Tab}|)\cdot|\mathrm{Tab}|$. The total number of reachable states is then bounded by $(|\mathrm{States}| + |\mathrm{Tab}|)\cdot(1 + (|\mathrm{States}| + |\mathrm{Tab}|)\cdot|\mathrm{Tab}|)$. □

The next two lemmas state the correctness of our construction, establishing a correspondence between winning strategies in $\mathcal{H}$ and Nash equilibria in $\mathcal{G}$.

[5] We explain in Theorem 30 how we can represent this winning condition in polynomial space, by using Boolean circuits.

Lemma 28. *Let $(v^A)_{A\in\mathrm{Agt}}$ be a payoff vector, and ρ be an infinite path in $\mathcal{G}$. The following two conditions are equivalent:*

- *player A_1 has a winning strategy in $\mathcal{H}(\mathcal{G},(v^A)_{A\in\mathrm{Agt}})$ from (s,Agt), and its outcome ρ' from (s,Agt) when A_2 obeys A_1 is such that $\pi_1(\rho')=\rho$;*
- *there is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$ in $\mathcal{G}$ from state s whose outcome from s is ρ.*

Proof. Assume there is a winning strategy σ^1 for player A_1 in $\mathcal{H}$ from (s,Agt). We define the strategy profile σ_{Agt} according to the actions played by A_1. Pick a history $g=s_1\cdot s_2\cdots s_{k+1}$, with $s_1=s$. Let h be the outcome of σ^1 from s ending in a player A_1 state and such that $\pi_1(h)=s_1\cdots s_k$. This history is uniquely defined as follows: the first state of h is (s_1,Agt), and if its $2i+1$-st state is (s_i,P_i), then its $2i+2$-nd state is $(s_i,P_i,\sigma^1(h_{\leq 2i+1}))$ and its $2i+3$-rd state is $(s_{i+1},P_i\cap\mathrm{Susp}((s_i,s_{i+1}),\sigma^1(h_{\leq 2i+1})))$. Now, write (s_k,P_k) for the last state of h, and let $h'=h\cdot(s_k,P_k,\sigma^1(h))\cdot(s_{k+1},P_k\cap\mathrm{Susp}((s_k,s_{k+1}),\sigma^1(h)))$. Then we define $\sigma_{\mathrm{Agt}}(g)=\sigma^1(h')$. Notice that when $g\cdot s$ is a prefix of $\pi_1(\rho')$ (where ρ' is the outcome of σ^1 from s when A_2 obeys A_1), then $g\cdot s\cdot\sigma_{\mathrm{Agt}}(g\cdot s)$ is also a prefix of $\pi_1(\rho')$.

We now prove that σ_{Agt} is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$. Pick a player $A\in\mathrm{Agt}$, a strategy σ'_A for A, and an infinite play g in $\mathrm{Out}(s,\sigma_{\mathrm{Agt}}[A\mapsto\sigma'_A])$. With g, we associate an infinite play h in $\mathcal{H}$ in the same way as above. Then player A is a suspect along all the transitions of g, so that she belongs to $L(h)$. Now, as σ^1 is winning, the payoff for A of $g=\pi_1(h)$ is less than v^A, which proves that σ_{Agt} is a trigger strategy.

Conversely, assume that σ_{Agt} is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$, and define the strategy σ^1 by $\sigma^1(h)=\sigma_{\mathrm{Agt}}(\pi_1(h))$. Notice that the outcome ρ' of σ^1 when A_2 obeys A_1 satisfies $\pi_1(\rho')=\rho$.

Let η be an outcome of σ^1 from s, and $A\in L(\eta)$. Then A is a suspect for each transition along $\pi_1(\eta)$, which means that for all i, there is a move m_i^A such that

$$\pi_1(\eta)_{=i+1}=\mathrm{Tab}(\pi_1(\eta)_{=i},\sigma_{\mathrm{Agt}}(\pi_1(\eta)_{\leq i})[A\mapsto m_i^A]).$$

Therefore there is a strategy σ'_A such that $\pi_1(\eta)=\mathrm{Out}(s,\sigma_{\mathrm{Agt}}[A\mapsto\sigma'_A])$. Since σ_{Agt} is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$, the payoff for player A of $\pi_1(\eta)$ is less than v_A. As this holds for any $A\in L(\eta)$, σ^1 is winning. □

Lemma 29. *Let ρ be an infinite path in $\mathcal{G}$ with payoff $(v^A)_{A\in\mathrm{Agt}}$. The following two conditions are equivalent:*

- *there is a path ρ' from (s,Agt) in $\mathcal{H}(\mathcal{G},(v^A)_{A\in\mathrm{Agt}})$ that never gets out of the winning region W of A_1, and along which A_2 always obeys A_1, and such that $\pi_1(\rho')=\rho$;*
- *there is a Nash equilibrium σ_{Agt} from s in $\mathcal{G}$ whose outcome is ρ.*

Proof. Let ρ be a path in $W(\mathcal{G},(v^A)_{A\in\mathrm{Agt}})$. We define a strategy σ^1 that follows ρ when A_2 obeys. Along ρ, this strategy is defined as follows: $\sigma^1(\rho_{\leq 2i})=m_{\mathrm{Agt}}$

such that $\mathrm{Tab}(\pi_1(\rho)_{=i}), m_{\mathrm{Agt}}) = \pi_1(\rho)_{=i+1})$. Such a legal move must exist since A_2 obeys A_1 along ρ. Now, if player A_2 deviates from the obeying strategy, we make σ^1 follow a winning strategy of A_1: given a finite outcome η of σ^1 that is not a prefix of ρ, we let j be the largest index such that $\eta_{\leq j}$ is a prefix of ρ. In particular, $\eta_{=j}$ belongs to the winning region W of A_1, and belongs to player A_2 (otherwise $\eta_{\leq j+1}$ would also be a prefix of ρ). Hence all the successors of $\eta_{=j}$ are in W. Thus player A_1 has a winning strategy $\hat{\sigma}^1$ from $\eta_{=j+1}$. We then define $\sigma^1(\eta_{\leq j} \cdot \eta') = \hat{\sigma}^1(\eta')$ for any outcome η' of $\hat{\sigma}^1$ from $\eta_{=j+1}$.

The outcomes of σ^1 are then either the path ρ, or a path obtained by following a winning strategy from some point on. Since $\mathcal{H}$ has a Muller winning condition, it follows that σ^1 is winning. Applying Lemma 28, we obtain a strategy profile σ_{Agt} in $\mathcal{G}$ that is a trigger strategy for $(v^A)_{A\in\mathrm{Agt}}$. Moreover, the outcome of σ_{Agt} from s in $\pi_1(\rho)$, so that σ_{Agt} is a Nash equilibrium.

Conversely, the Nash equilibrium is a trigger strategy, and from Lemma 28, we get a winning strategy σ^1 in $\mathcal{H}$. The outcome ρ of σ^1 from s when A_2 obeys A_1 is such that $\pi_1(\rho)$ is the outcome of the Nash equilibrium, so that its payoff is $(v^A)_{A\in\mathrm{Agt}}$. Since σ^1 is winning, ρ never gets out of the winning region, which concludes the proof. □

6.2 General case

As noticed in [10], the algorithm from [14] to compute the winning states in a game can be adapted to the case where the winning conditions are given as a Boolean circuit (the circuit has as many input gates as the number of states, and a path is declared winning if the circuit evaluates to 1 when setting the input gates to 1 for the states that are visited infinitely often). This algorithm uses polynomial space.

Theorem 30. *For Büchi objectives with preorders given as Boolean circuits, the value, existence and constrained existence problems are* PSPACE*-complete.*

Proof. The algorithm proceeds by trying all the possible payoffs $(v^A)_{A\in\mathrm{Agt}}$. We consider the game $\mathcal{H}$; in that game, the objective for player A_1 can still be defined as a Boolean circuit of polynomial size, by combining the circuits of the players in Agt. We then use the polynomial-space algorithm from [10] to compute the winning regions in this game. Finally, we look for a path in $\mathcal{H}$ as described in Lemma 29; this again is in polynomial space as we can restrict to plays of the form $\pi \cdot \tau^\omega$ with $|\pi| \leq |\mathrm{States}|^2$ and $|\tau| \leq |\mathrm{States}|^2$ (Lemma 5).

PSPACE-hardness for the value problem was already proven in [10]. For the existence problem, given a two-player game $\mathcal{G}$ and a state s, we define a new game $\mathcal{G}'$ in which both players begin with playing a matching-penny game, and either go to the initial state s of $\mathcal{G}$ (and then play in $\mathcal{G}$), or to go to a final state that is winning for A_1 and losing for A_2. The objective for A_2 in $\mathcal{G}$ is set to be the opposite of that of A_1. If there is a Nash equilibrium in $\mathcal{G}'$, then player A_1 must be winning in both part of $\mathcal{G}'$, which means she has a winning strategy in $\mathcal{G}$. Conversely, if she has a winning strategy in $\mathcal{G}$, then it can be combined with any strategy of A_2 to give a Nash equilibrium in $\mathcal{G}'$. ◻

6.3 Simple cases

Reduction to a single Büchi objective. The preorders that were reducible to a single reachability objectives in the case of reachability can also be reduced to a single Büchi objectives in the Büchi case: just replace Occ with Inf. The same holds of co-reducibility. The algorithm in [3] can then be adapted to solve the problem.

Proposition 31. *We consider Büchi objectives. For a monotonic preorder that is reducible to a single objective, the value problem is* P*-complete. For a preorder that is co-reducible to a single objective, existence and constrained existence are* P*-complete.*

We prove each item of this proposition in the following lemmas.

Lemma 32. *For a monotonic preorder that is reducible to a single (Büchi) objective, the value problem is in* P*.*

Proof. Just transform the objectives into a single one, and use a polynomial-time algorithm [14,4] to solve the resulting Büchi game. ◻

For the constrained existence problem, we adapt the set-based characterization for Nash equilibria of [3]. Consider constraints given as vectors u^A and w^A for each player A, and an initial state s. For each $A \in \mathrm{Agt}$, we write $v^A(K) = \mathbb{1}_{\{i \mid K \cap T_i^A \neq \varnothing\}}$, and look for a transition system $\langle K, E\rangle$, with $K \subseteq \mathrm{States}$, for which the following properties hold:

(1) $u^A \leq v^A(K) \leq w^A$ for all $A \in \mathrm{Agt}$;
(2) $\langle K, E\rangle$ is strongly connected;
(3) $\forall k \in K.\ (k, \mathrm{Agt}) \in W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$;
(4) $\forall (k, k') \in E.\ \exists (k, \mathrm{Agt}, m_{\mathrm{Agt}}) \in W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}}).\ \mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'$;
(5) $(K \times \{\mathrm{Agt}\})$ is reachable from (s, Agt) in $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$;

where $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ is the winning region of A_1 in $\mathcal{H}(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$.

Lemma 33. *There is a transition system $\langle K, E\rangle$ satisfying conditions (1)–(5) iff there is a path ρ from (s, Agt) that never gets out of $W(\mathcal{G}, (v^A(\mathrm{Inf}(\rho)))_{A\in\mathrm{Agt}})$, along which A_2 always obeys A_1, and $u^A \leq v^A(\mathrm{Inf}(\rho)) \leq w^A$ for all $A \in \mathrm{Agt}$.*

Proof. The first implication is shown by building a path in $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ that successively visits all the states in $K \times \{\mathrm{Agt}\}$ forever. Thanks to (5), (2) and (4) (and the fact that A_2 obeys A_1), such a path exists, and from (3) and (4), this path remains in the winning region. From (1), we have the condition on the payoff. Reciprocally, consider such a path ρ, and let $K = \pi_1(\mathrm{Inf}(\rho) \cap V_1)$ and $E = \{(k, k') \in K^2 \mid \exists (k, \mathrm{Agt}, m_{\mathrm{Agt}}) \in \mathrm{Inf}(\rho).\ \mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'\}$. Condition (5) clearly holds. Conditions (1), (3) and (4) are easy consequences of the hypotheses and construction. We prove that $\langle K, E\rangle$ is strongly connected. First, since A_2 obeys A_1 and ρ starts in (k, Agt), we have $L(\rho) = \mathrm{Agt}$. Now, take any two states k and k' in K: then ρ visits (k, Agt) and (k', Agt) infinitely often, and there is a subpath of ρ between those two states, all of which states appear infinitely often along ρ. Such a subpath gives rise to a path between k and k', as required. $\square$

As a consequence, if $\langle K, E\rangle$ satisfies the conditions, by Lemma 29, there is a Nash equilibrium with payoff $(v^A(K))_{A\in\mathrm{Agt}}$. Our aim is to compute in polynomial time all maximal pairs $\langle K, E\rangle$ that satisfy the condition. We first need to compute $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$, given $(v^A(K))_{A\in\mathrm{Agt}}$.

Lemma 34. *The set $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ can be computed in polynomial time.*

Proof. We use the fact that $\mathcal{H}(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ is a (turn-based) co-Büchi game. Indeed, since each player has a single Büchi objective in $\mathcal{G}$, player A_1 has to enforce finitely many visits to the states (s, P) where $s \in \Omega^A_1$ and $A \in P$. The winning region can then be determined using the polynomial time algorithm of Lemma 32 for Büchi games. $\square$

Now, we define a recursive function SSG (standing for "solve sub-game"), working on transition systems:

- if $K \times \{\mathrm{Agt}\} \subseteq W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$, and for all $(k, k') \in E$, there is a $(k, \mathrm{Agt}, m_{\mathrm{Agt}})$ in $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ s.t. $\mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'$, and finally $\langle K, E\rangle$ is strongly connected, then we set $\mathsf{SSG}(\langle K, E\rangle) = \{\langle K, E\rangle\}$;
- otherwise, we let

  $$\mathsf{SSG}(\langle K, E\rangle) = \bigcup_{\langle K', E'\rangle \in \mathrm{SCC}(\langle K, E\rangle)} \mathsf{SSG}(T(\langle K', E'\rangle))$$

  where $\mathrm{SCC}(\langle K, E\rangle)$ is the set of strongly connected components of $\langle K, E\rangle$ (which can be computed in linear time), and where $T(\langle K', E'\rangle)$ is the transition system whose set of states is $\{k \in K' \mid (k, \mathrm{Agt}) \in W(\mathcal{G}, (v^A(K'))_{A\in\mathrm{Agt}})\}$ and whose set of edges is

  $$\{(k, k') \in E' \mid \exists (k, \mathrm{Agt}, m_{\mathrm{Agt}}) \in W(\mathcal{G}, (v^A(K'))_{A\in\mathrm{Agt}}).\ \mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'\}.$$

  Notice that this set of edges is never empty, but $T(\langle K', E'\rangle)$ might not be strongly connected anymore, so that this is really a recursive definition.

For each player A, let T'_A be the target associated with the upper-bound w^A when we "co-reduce" it to a single reachability objective. We have to ensure that it is not in K, so that the payoff does not exceed w^A. We define

$$\mathsf{Sol} = \mathsf{SSG}\big(\langle \mathrm{States} \setminus \cup_{A\in\mathrm{Agt}} T'_A, \mathrm{Edg}\rangle\big) \cap \big\{\langle K, E\rangle \mid \forall A \in \mathrm{Agt}.\ u^A \lesssim v^A(K)\big\}$$

Lemma 35. *If $\langle K, E\rangle \in$ Sol then it satisfies conditions* (1) *to* (4)*. Conversely, if $\langle K, E\rangle$ satisfies conditions* (1) *to* (4)*, then there exists $\langle K', E'\rangle \in$ Sol such that $\langle K, E\rangle \subseteq \langle K', E'\rangle$.*

Proof. Let $\langle K, E\rangle \in \mathsf{Sol}$. By definition of SSG, all (k, Agt) for $k \in K$ are in $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$, and for all $(k, k') \in E$, there is a state $(k, \mathrm{Agt}, m_{\mathrm{Agt}})$ in $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ such that $\mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'$, and $\langle K, E\rangle$ is strongly connected. Also, for all A, $u^A \lesssim v^A(K)$ because $\mathsf{Sol} \subseteq \{\langle K, E\rangle \mid u^A \lesssim v^A(K)\}$. Finally, for any $A \in \mathrm{Agt}$, $v^A(K) \lesssim w^A$ because the set K does not contain any state in T'_A.

Conversely, assume that $\langle K, E\rangle$ satisfies the conditions. We show that if $\langle K, E\rangle \subseteq \langle K', E'\rangle$ then there is $\langle K'', E''\rangle$ in $\mathsf{SSG}(\langle K', E'\rangle)$ such that $\langle K, E\rangle \subseteq \langle K'', E''\rangle$. The proof is by induction on the size of $\langle K', E'\rangle$.

The basic case is when $\langle K', E'\rangle$ satisfies conditions (2), (3), and (4): then $\mathsf{SSG}(\langle K', E'\rangle) = \{\langle K', E'\rangle\}$, and by letting $\langle K'', E''\rangle = \langle K', E'\rangle$ we get the expected result.

We now analyze the other case. There is a strongly connected component of $\langle K', E'\rangle$, say $\langle K'', E''\rangle$, which contains $\langle K, E\rangle$, because $\langle K, E\rangle$ satisfies condition (2). We have $v^A(K) \lesssim v^A(K'')$ (because $K \subseteq K''$) for every A, and thus $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}}) \subseteq W(\mathcal{G}, (v^A(K''))_{A\in\mathrm{Agt}})$. This ensures that $T(\langle K'', E''\rangle)$ contains $\langle K, E\rangle$ as a subgraph. Since $\langle K'', E''\rangle$ is a subgraph of $\langle K', E'\rangle$, the graph $T(\langle K'', E''\rangle)$ also is. We show that they are not equal, so that we can apply the induction hypothesis to $T(\langle K'', E''\rangle)$. For this, we exploit the fact that $\langle K', E'\rangle$ does not satisfy one of conditions (2) to (4):

- first, if $\langle K', E'\rangle$ is not strongly connected while $\langle K'', E''\rangle$ is, they cannot be equal;
- if there is some $k \in K'$ such that (k, Agt) is not in $W(\mathcal{G}, (v^A(K'))_{A\in\mathrm{Agt}})$, then k is not a vertex of $T(\langle K'', E''\rangle)$;
- if there some edge (k, k') in E' such that there is no state $(k, \mathrm{Agt}, m_{\mathrm{Agt}})$ in $W(\mathcal{G}, (v^A(K'))_{A\in\mathrm{Agt}})$ such that $\mathrm{Tab}(k, m_{\mathrm{Agt}}) = k'$, then the edge (k, k') is not in $T(\langle K'', E''\rangle)$.

We then apply the induction hypothesis to $T(\langle K'', E''\rangle)$, and get the expected result. Now, because of condition (1), $u^A \lesssim v^A(K) \lesssim w^A$. Hence, due to the previous analysis, there exists $\langle K', E'\rangle \in \mathsf{SSG}(\langle \mathrm{States} \setminus \cup_{A\in\mathrm{Agt}} T'_A, \mathrm{Edg}\rangle)$ such that $\langle K, E\rangle \subseteq \langle K', E'\rangle$. This concludes the proof of the lemma. □

Lemma 36. *The set Sol can be computed in polynomial time.*

Proof. Each recursive call to SSG applies to a decomposition in strongly connected components of the current transition system under consideration. Hence the number of recursive calls is bounded by $|\mathrm{States}|^2$. Computing the decomposition in SCCs can be done in linear time. Furthermore, Lemma 34 shows that $W(\mathcal{G}, (v^A(K))_{A\in\mathrm{Agt}})$ can be computed in polynomial time. Hence globally we can compute Sol in polynomial time. □

Lemma 37. *For Büchi objectives and a preorder that is co-reducible to a single objective, the constrained existence problem is in* P*.*

Proof. The algorithm computes Sol in polynomial time, then looks for a path from (s, Agt) that never gets out of the winning region, so that condition (5) is satisfied. Thanks to the characterization in Lemma 33, this is correct and complete. $\square$

Reduction to a deterministic Büchi automaton. For some preorders, given any payoff u, it is possible to construct (in polynomial time) a deterministic Büchi automaton that recognizes the plays whose payoff v for player A is higher than u (i.e. $u \lesssim v$). When this is the case, we say that the preorder is *reducible to a deterministic Büchi automaton*.

Proposition 38. *For Büchi objectives with a preorder reducible to a deterministic Büchi automaton, the value problem is in* P*. In particular it is* P*-complete for conjunction, lexicographic and subset preorders.*

Proof. Given the payoff v^A for player A, the algorithm proceeds by constructing the automaton that recognizes the plays with payoff higher than v^A. By performing the product with the game as described in [3], we obtain a new game, where there is a winning strategy if and only if there is a strategy in the original game to ensure payoff v^A. In this new game, A has a single Büchi objective, so that the existence of a winning strategy can be decided this in polynomial time.

Hardness in Palready holds for games with a single Büchi objective. $\square$

Lemma 39. *The conjunction preorder is reducible to a deterministic Büchi automaton.*

Proof. Let $(T_i^A)_{1 \le i \le n}$ be the targets defining the partial objectives of player A. We construct a deterministic Büchi automaton $\mathcal{A}_A$ to describe the objectives of A. There is one state for each target, plus one accepting state: $\mathcal{Q}_A = \{q_0, q_1, \ldots, q_n\}$, and $R_A = \{q_0\}$. For all $i \in [1, n]$, the transitions are $q_{i-1} \xrightarrow{s} q_i$ if $s \in T_i^A$ and $q_i \xrightarrow{s} q_i$ otherwise, we also have a transition $q_n \xrightarrow{\mathrm{States}} q_0$. This automaton describe the paths that goes through each T_i^A infinitely often, its size is $n + 1$. $\square$

Lemma 40. *The subset order is reducible to a deterministic Büchi automaton.*

Proof. If the targets are $(T_i^A)_{1 \le i \le n}$ and the payoff is u, then the plays with payoff v such that $u \lesssim v$ are those that visit all the targets T_i^A with $u_i = 1$; this is equivalent to the conjunction where we only keep these targets. Therefore we can use a construction similar to the one in the previous proof. $\square$

Lemma 41. *The lexicographic order is reducible to a deterministic Büchi automaton.*

Proof. Assume the targets of player A are given by $(T_i^A)_{1 \le i \le n}$. Given a payoff vector u in $\{0, 1\}^n$, we construct the following automaton which recognizes the runs whose value is greater than or equal to u. In this automaton there is a state q_i for each u_i that is equal to 1 and a state q_0 that is both initial and final: $Q = \{q_i \mid u_i = 1\} \cup \{q_0\}$; $I = \{q_0\}$; $R = \{q_0\}$. The transition function is defined as follows. From q_0 the run always go to q_i: $q_0 \xrightarrow{s} q_i$ where i is the smallest integer such that $u_i = 1$. From any state q_i with $i > 0$,

- if the automaton receives T_i^A as input (i.e., if a state in T_i^A is visited in the game), then the automaton jumps to the next state q_{i+j} (or to q_0 if q_i is the last state of Q);
- if the automaton receives T_k^A as input, with $u_k = 0$ and $k < i$, then the automaton goes to q_0;
- in the other cases, the automaton loops in q_i.

An example of the construction is given in Fig. 8.

We now prove correctness of this construction. Consider a path that goes from q_0 to q_0: if the automaton is currently in state q_i, then since the last occurrence of q_0, at least one state for each target T_j^A such that $j < i$ and $u_j = 1$ has been visited. When q_0 is reached again, either it is because we have seen all the T_j^A with $u_j = 1$, or it is because the run visited some target T_i^A with $u_i = 0$ and all the T_j^A such that $u_j = 1$ and $j < i$; in both case case, the set of targets that have been visited between two visits to q_0 describe a payoff greater than u. Assume the word π is accepted by the automaton; then there is a sequence of q_i as above that is taken infinitely often, therefore the vector $v = \mathbb{1}_{\{j \mid T_j^A \cap \mathrm{Inf}(\pi) \neq \varnothing\}}$ is greater than or equal to u for the lexicographic order.

Reciprocally assume $v = \mathbb{1}_{\{j \mid T_j^A \cap \mathrm{Inf}(\pi) \neq \varnothing\}}$ is greater than or equal to u, that we already read a prefix $\pi_{\leq k}$ for some k, and that the current state is q_0. Reading the first symbol in π after position k, the run goes to the state q_i where i is the least integer such that $u_i = 1$. Either the path visits T_i^A at some point, or it visits a state in a target T_j^A, with j smaller than i and $v_j = 0$, in which case the automaton goes to q_0. Therefore from q_0 we can again come back to q_0 while reading the following of π, and the automaton accepts. $\square$

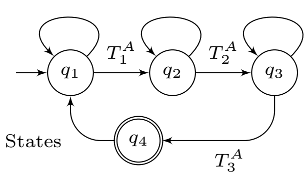


Fig. 7. The automaton for the conjunction preorder, $n = 3$

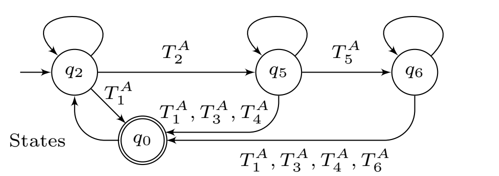


Fig. 8. The automaton for the lexicographic order, $n = 7$ and $u = (0, 1, 0, 0, 1, 1, 0)$

Using the same techniques, we now consider the parity preorder with *reachability* objectives. Notice that the parity preorder is not monotonic, so that when associated with reachability objectives, it imposes both reachability and safety objectives. This is why a Büchi condition can help for this case. We thus recover the complexity result about weak parity games from [16].

Lemma 42. *For* reachability *objectives, the parity preorder is reducible to a deterministic Büchi automaton.*

Proof. We construct an automaton $\mathcal{A}$ that remembers the highest parity that have been seen so far. Assume player A has n target states, the states of the automaton are $Q = \{q_0, q_1 \dots q_n\}$, the transition function is defined by $\delta\colon (q_i, s) \to q_j$ where j is $\max\{i\} \cup \{k \mid s \in T_k^A\}$. The final states are the q_i whose index i are even. The automaton accepts exactly the plays whose highest visited target as an even index. The construction is illustrated in Fig. 9. $\square$

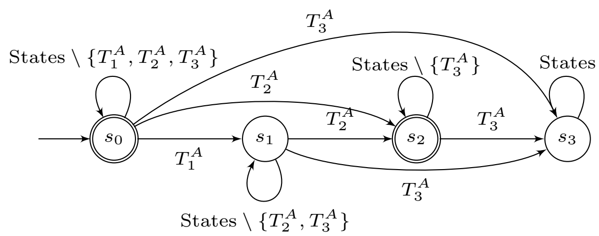


Fig. 9. Automaton for the parity preorder and three reachability objectives

6.4 Monotonic preorders

We say that a strategy profile is *memoryless* if there exists a function $f\colon \text{States} \to \text{Act}^{\text{Agt}}$ such that $\sigma_{\text{Agt}}(\rho{\cdot}s) = f(s)$. We show that when the preorder is monotonic, our problems are also easier than in the general case. This is because we can find memoryless trigger strategies.

Lemma 43. *Let $\mathcal{H}$ be a two-player game, σ_1 be a strategy for Player* 1*, and s_0 be a state of $\mathcal{H}$. There is a memoryless strategy σ_1' such that for all outcome ρ' of σ_1' from s_0, there is an outcome ρ of σ_1 from s_0 such that* $\text{Inf}(\rho') \subseteq \text{Inf}(\rho)$.

Proof. This proof is by induction on the size of the set $S(\sigma_1) = \{(s, m) \mid \exists h.\ \sigma_1(h \cdot s) = m\}$. If its size is the same as States then the strategy is memoryless. Otherwise, let s be a state where σ_1 takes several different actions (i.e., $|(\{s\} \times \text{Act}) \cap S(\sigma_1)| > 1$), and m be an action such that $(s, m) \in S(\sigma_1)$.

We will define a new strategy σ_1' that takes fewer different actions in s and such that for any outcome of σ_1' there is an outcome of σ_1 that visits (at least) the same states infinitely often.

If σ is a strategy and h a history, we let $\sigma \circ h\colon h' \mapsto \sigma(h \cdot h')$ for any history h'.

If there is a history h_0 compatible with σ_1 such that the action m is never taken again by σ_1, then we define $\sigma_1' = \sigma_1 \circ h$. It is clear that all the states visited by σ_1' are visited by σ_1 and we removed one action from the strategy: $S(\sigma_1') \subseteq S(\sigma_1) \setminus \{(s,m)\}$.

Otherwise, we can extend any finite outcome of σ_1 by one where σ_1 takes m in state s. Let h be an history, we define the extension $e(h)$ inductively in that way:

- $e(\varepsilon) = \varepsilon$, where ε is the empty history;
- $e(h \cdot s) = e(h) \cdot h'$ where h' ends with an s, is an outcome of $\sigma_1 \circ e(h)$ and $\sigma_1(e(h) \cdot h') = m$;
- $e(h \cdot s') = e(h) \cdot s'$ if $s' \neq s$.

We extend the definition of e to infinite outcomes in the natural way: $e(\rho)_i = e(\rho_{\leq i})_i$, and define the strategy $\sigma_1' \colon h \mapsto \sigma_1(e(h))$.

Assume h is a finite outcome of σ_1', that $e(h)$ is an outcome of σ_1 and $\mathrm{last}(h) = \mathrm{last}(e(h))$. If $h \cdot s$ is an outcome of σ_1', by construction of e, $e(h \cdot s) = e(h) \cdot h'$, such that $\mathrm{last}(h') = s$, and h' is an outcome of $\sigma_1 \circ e(h)$ and as $e(h)$ is an outcome of σ_1 by hypothesis, that means that $e(h \cdot s)$ is an outcome of σ_1. If $h \cdot s'$ with $s' \neq s$ is an outcome of σ_1', $e(h \cdot s') = e(h) \cdot s'$, $s' \in \mathrm{Tab}(\mathrm{last}(h), \sigma_1'(h))$, $\sigma_1'(h) = \sigma_1(e(h))$, using the hypothesis $\mathrm{last}(h) = \mathrm{last}(e(h))$, and $e(h)$ is an outcome of σ_1, therefore $e(h \cdot s')$ is an outcome of σ_1. This shows that if ρ is a finite outcome of σ_1' then $e(\rho)$ is an outcome of σ_1. Notice furthermore that σ_1' in s only takes the action m.

We then have $S(\sigma_1') < S(\sigma_1)$, and the induction hypothesis entails the result. □

Lemma 44. *For Büchi objectives with a monotonic preorder, if there is a trigger strategy for $(v^A)_{A \in \mathrm{Agt}}$ from s, then there is a memoryless winning strategy for A_1 in $\mathcal{H}(\mathcal{G}, (v^A)_{A \in \mathrm{Agt}})$ from state (s, Agt).*

Proof. If there is a trigger strategy for $(v^A)_{A \in \mathrm{Agt}}$ then, because of Lemma 28, there is a winning strategy σ^1 in game $\mathcal{H}(\mathcal{G}, (v^A)_{A \in \mathrm{Agt}})$ for player A_1. Consider the memoryless strategy σ_1' constructed as in Lemma 43. Let ρ' be an outcome of σ_1', there is an outcome ρ of σ_1 such that $\mathrm{Inf}(\rho') \subseteq \mathrm{Inf}(\rho)$. For each player A, $\{i \mid \mathrm{Inf}(\rho') \cap T_i^A\} \subseteq \{i \mid \mathrm{Inf}(\rho) \cap T_i^A\}$, as the preorder is monotonic the payoff of ρ' is smaller that that of ρ. So the play is winning for A and σ_1' is a memoryless winning strategy. □

Lemma 45. *For Büchi objectives with a preorder defined by a monotonic circuit, given a set P of players, we can decide in polynomial time if a memoryless strategy for P in $\mathcal{H}(\mathcal{G}, (v^A)_{A \in \mathrm{Agt}})$ is winning.*

Proof. Let σ^1 be a memoryless strategy in $\mathcal{H}(\mathcal{G}, (v^A)_{A \in \mathrm{Agt}})$. By keeping only the edges that are taken by σ^1, we define a subgraph of the game. We compute the strongly connected component of this graph. If one component is reachable and does not satisfy the objective of A_1, then the strategy is not winning. Reciprocally if all the reachable strongly connected components satisfies the winning condition of A_1 and because the preorder is monotonic, σ_1 is winning. □

Proposition 46. *For Büchi objectives with preorders given by monotonic circuits, the value problem is* coNP*-complete and the existence and constrained existence problem are* NP*-complete. For the counting preorder the value problem is* coNP*-complete, and the existence and constrained existence are* NP*-complete. For monotonic preorders with an element v such that $u \not\lesssim v \Leftrightarrow u = \mathbb{1}$, the constrained existence problem is* NP*-complete.*

Proof. For the value problem, we can make the concurrent game turn-based: since A_1 must win against any strategy of the opponent, she must also win in the case where the opponents' strategies can adapt to what A_1 plays. This turn-based game is determined, so that there is a strategy σ^1 whose outcomes are always better (for A_1) than v^{A_1} if and only if, for any strategy σ^2 of player A_2, there is an outcome with payoff (for A_1) better than v^{A_1}. If there is a counterexample to this fact, then thanks to Lemma 43 there is one with a memoryless strategy σ^2. The coNP algorithm proceeds by checking that all the memoryless strategies of player A_2 have an outcome better than v^{A_1}, which is achievable in polynomial time, with a method similar to Lemma 45.

The algorithm for the constrained existence problem proceeds by guessing

- the payoff for each player,
- a play of the form $\pi \cdot \tau^\omega$, where $|\pi| \leq |\text{States}|^2$ and $|\tau| \leq |\text{States}|^2$,
- an under-approximation of the set of winning states in $\mathcal{H}(\mathcal{G}, (v^A)_{A\in\text{Agt}})$, where v^A is the payoff of $\pi \cdot \tau^\omega$ for player A,
- a memoryless strategy profile σ_{Agt}.

We check that σ_{Agt} is a witness for the fact that the state in the under-approximation are winning; thanks to Lemma 45, verifying this can be checked in polynomial time. We also verify that the play has the expected payoff, that the payoff satisfies the constraints, and that it never gets out of the under-approximation of the set of winning states. If these conditions are fulfilled, then the play meet the conditions of Lemma 29, and there is a Nash equilibrium. Lemmas 44 and 5 ensure that if there is a Nash equilibrium, we can find it this way.

We now turn to the hardness proofs.

Lemma 47. *For Büchi objectives with the counting preorder, the value problem is co-*NP*-hard.*

Proof. We reduce (the complement of) 3SAT into the value problem for two-player turn-based games with Büchi objectives with the counting preorder. Consider an instance

$$\phi = C_1 \wedge \cdots \wedge C_m$$

with $C_j = \ell_{j,1} \vee \ell_{j,2} \vee \ell_{j,3}$, over a set of variables $\{x_1, \ldots, x_n\}$. With ϕ, we associate a two-player turn-based game $\mathcal{G}$. Its set of states is made of

- a set containing the unique initial state $V_0 = \{s_0\}$,
- a set of two states $V_k = \{x_k, \neg x_k\}$ for each $1 \leq k \leq n$,
- and a set of three states $V_{n+j} = \{t_{j,1}, t_{j,2}, t_{j,3}\}$ for each $1 \leq j \leq m$.

Then, for each $0 \le l \le n+m$, there is a transition between any state of V_l and any state of V_{l+1} (assuming $V_{n+m+1} = V_0$). The game involves two players: player B owns all the states, but has no objectives (she always loses). Player A has a set of Büchi objectives defined by $T^A_{2\cdot k} = \{x_k\} \cup \{t_{j,p} \mid \ell_{j,p} = x_k\}$, $T^A_{2\cdot k+1} = \{\neg x_k\} \cup \{t_{j,p} \mid \ell_{j,p} = \neg x_k\}$, for $1 \le k \le n$. Notice that at least n of these objectives will be visited infinitely often along any infinite play. We prove that if the formula is not satisfiable, then at least $n+1$ objectives will be fulfilled, and conversely.

Assume the formula is satisfiable, and pick a witnessing valuation v. We define a strategy σ_B for B that "follows" valuation v: from states in V_{k-1}, for any $1 \le k \le n$, the strategy plays towards x_k if $v(x_k) = \texttt{true}$ (and to $\neg x_k$ otherwise). Then, from a state in V_{n+l-1} with $1 \le l \le m$, it plays towards one of the $t_{j,p}$ that evaluates to true under v (the one with least index p, say). This way, the number of targets of player A that are visited infinitely often is n.

Conversely, pick a play in $\mathcal{G}$ s.t. at most (hence exactly) n objectives of A are fulfilled. In particular, for any $1 \le k \le n$, this play never visits one of x_k and $\neg x_k$, so that it defines a valuation v over $\{x_1, \ldots, x_n\}$. Moreover, any state of V_{n+l}, with $1 \le l \le p$, that is visited infinitely often must correspond to a literal that is made true by v, as otherwise this would make one more objective that is fulfilled for A. As a consequence, each clause of ϕ evaluates to true under v, and the result follows. □

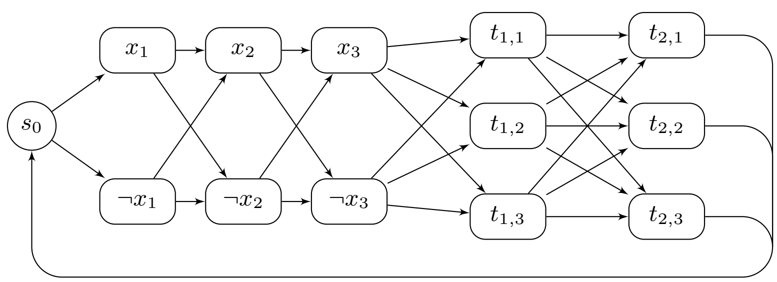


Fig. 10. The game $\mathcal{G}$ associated with formula ϕ of (2)

Example 48. As an example, we illustrate the construction in Fig. 10 for the formula

$$\varphi = (x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee \neg x_3)\,. \tag{2}$$

Lemma 49. *For Büchi objectives with the counting preorder, the existence problem is* NP*-hard.*

Proof. Let $\mathcal{G}$ be the game we constructed for Lemma 47. We add an initial state, in which players A and B play an instance of the matching-penny game, and either go to the game $\mathcal{G}$ or to a new state z_0 that satisfies n objectives of A, and is winning for player B. If the value of A is $n+1$ in $\mathcal{G}$ then there is no equilibrium. Conversely, if the value of A is not $n+1$ in $\mathcal{G}$, then B has a strategy that ensures that A does not satisfies more than n of her objectives, and going to z_0 is a Nash equilibrium. $\square$

Lemma 50. *For Büchi objectives, a monotonic preorder with an element v^n for all n, such that $u \neq \mathbb{1}$ if and only if $u \lesssim v$, the constrained existence problem for turn-based games is* NP*-hard.*

Proof. Let us consider a formula $\phi = C_1 \wedge \cdots \wedge C_m$ We use the same construction as in the proof of Lemma 11, but with no "universal state". For each variable x_i, our game has one player B_i and three states s_i, x_i and $\neg x_i$. The objectives of B_i are the sets $\{x_i\}$ and $\{\neg x_i\}$. Transitions go from each s_i to x_i and $\neg x_i$, and from x_i and $\neg x_i$ to s_{i+1} (with $s_{n+1} = s_0$). Finally, an extra player A has full control of the game (i.e., she owns all the states) and has n objectives, defined by $T_i^A = \{\ell_{i,1}, \ell_{i,2}, \ell_{i,3}\}$ for $1 \leq i \leq n$.

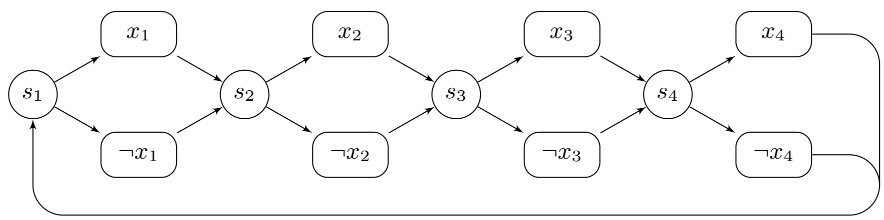


Fig. 11. The Büchi game for a formula with 4 variables

We show that formula ϕ is satisfiable if, and only if, there is a Nash equilibrium where each player B_i gets payoff β_i satisfying $\beta_i \lesssim v^2$ (hence $\beta_i \neq (1,1)$), and player A gets payoff $\mathbb{1}$.

First assume that the formula is satisfiable, and pick a witnessing valuation v. By playing according to v, player A can satisfy all of her objectives (hence she cannot improve her payoff, since the preorder is monotonic). Since she alone controls all the game, the other players cannot improve their payoff, so that this is a Nash equilibrium. Moreover, since A plays memoryless, only one of x_i and $\neg x_i$ is visited for each i, so that the payoff β_i for B_i satisfies $\beta_i \lesssim v^2$.

Conversely, if there is a Nash equilibrium with the desired payoff, then by hypothesis, exactly one of each x_i and $\neg x_i$ is visited infinitely often (so that the payoff for B_i is not $(1,1)$), which defines a valuation v. Since in this Nash equilibrium, player A satisfies all its objectives, one state of each target is visited, which means that under valuation v, formula ϕ evaluates to true. $\square$

$\square$

Applying Lemma 10, the result of Proposition 46 apply in particular to the conjunction and lexicographic preorders, for which the constrained existence problem is thus NP-complete.

6.5 Parity games

Proposition 51. *For Büchi objectives with the parity preorder, the constrained existence problem is* $\mathsf{P}_{\|}^{\mathsf{NP}}$*-complete.*

Proof. We first describe our $\mathsf{P}_{\|}^{\mathsf{NP}}$ algorithm.

To begin with, consider any set $L \subseteq \mathrm{Agt}$; with L, we associate a payoff tuple $\mathbf{v}(L) = (v^A)_{A\in\mathrm{Agt}}$ where $v^A = \mathbb{1}_{\{1\}}$ (which is losing for the parity preorder) if A is in L and $v^A = \mathbb{1}_{\{2\}}$ if not. For the parity preorder, any payoff vector $(v^A)_{A\in\mathrm{Agt}}$ is equivalent to (i.e., both smaller than and greater than) $\mathbf{v}(L)$, where L is the set of players that are losing in $(v^A)_{A\in\mathrm{Agt}}$.

Our algorithm relies on the games $\mathcal{H}(\mathcal{G}, (v^A)_{A\in\mathrm{Agt}})$ defined in Section 6.1. First notice that given two payoff vectors $(v^A)_{A\in\mathrm{Agt}}$ and $(u^A)_{A\in\mathrm{Agt}}$, the games $\mathcal{H}(\mathcal{G}, (v^A)_{A\in\mathrm{Agt}})$ and $\mathcal{H}(\mathcal{G}, (u^A)_{A\in\mathrm{Agt}})$ only differ in their winning conditions. In particular, the structure of the game only depends on $\mathcal{G}$, and has polynomial size (see Lemma 27).

Moreover, given a payoff vector $(v^A)_{A\in\mathrm{Agt}}$, the winning condition for A_1 in $\mathcal{H}(\mathcal{G}, (v^A)_{A\in\mathrm{Agt}})$ is the same as in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$ where L is the set of losing players in $(v^A)_{A\in\mathrm{Agt}}$; this is because the parity preorder has only two equivalence classes. In the end, a path ρ is winning for A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$ iff each player in $L(\rho) \cap L$ loses.

Write $\mathcal{P}$ for the set of sets of players of Agt that appear as the second item of a state of $\mathcal{H}$:

$$\mathcal{P} = \{P \subseteq \mathrm{Agt} \mid \exists s \in \mathrm{States}.\ (s, P) \text{ is a state of } \mathcal{H}\}.$$

Since $\mathcal{H}$ has size polynomial, so does $\mathcal{P}$. Also, for any path ρ, $L(\rho)$ is a set of $\mathcal{P}$. Hence, for a fixed L, the number of sets $L(\rho)\cap L$ is polynomial. Now, the winning condition for A_1 is that the players in $L(\rho) \cap L$ must be losing. This means that for each set P that is a possible value for $L(\rho)\cap L$, if a path γ infinitely often has, as its second component, a set Q s.t. $Q \cap L = P$, then those players in P will not meet their objectives along γ. This can be encoded as a single parity objective, setting a high even priority for those sets not having a set Q s.t. $Q \cap L = P$ as their second element, and their priority obtained from the original game for the other states. The global winning condition for A_1 will then be the conjunction of those (polynomially-many) parity conditions.

Now, deciding whether a state is winning in a turn-based game for such generalized parity condition can be decided in coNP [5]. Hence, given a state $s \in \mathrm{States}$ and a payoff vector $\mathbf{v}(L)$, we can decide in coNP whether s is winning for A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$. This will be used as an oracle in our algorithm below.

Now, pick a set $P \subseteq \mathrm{Agt}$ of suspects, i.e., for which there exists $(s,t) \in \mathrm{States}^2$ and m_{Agt} s.t. $P = \mathrm{Susp}((s,t), m_{\mathrm{Agt}})$. Using the same arguments as in the proof of Lemma 27, it can be shown that $2^{|P|} \leq |\mathrm{Tab}|$, so that the number of subsets of P is polynomial. Now, for each set P of suspects and each $L \subseteq P$, write $w(L)$ for the size of the winning region of A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$. Then the sum $\sum_{P\in\mathcal{P}\setminus\{\mathrm{Agt}\}} \sum_{L\subseteq P} w(L)$ is at most $|\mathrm{States}| \times |\mathrm{Tab}|^2$.

Assume that the exact value M of this sum is known, and consider the following algorithm:

1. for each $P \subseteq \mathcal{P} \setminus \{\text{Agt}\}$ and each $L \subseteq P$, guess a set $W(L) \subseteq \text{States}$, which we intend to be the exact winning region for A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$.
2. check that the sizes of those sets sum up to M;
3. for each $s \notin W(L)$, check that A_1 does not have a winning strategy from s in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$. This can be checked in NP, as explained above.
4. guess a lasso-shaped path $\rho = \pi \cdot \tau^\omega$ in $\mathcal{H}$ starting from (s, Agt), with $|\pi|$ and $|\tau|$ less than $|\text{States}|^2$ (following Lemma 5) visiting only states where the second item is Agt. This path can be seen as the outcome of some strategy of A_1 when A_2 obeys. For this path, we then check the following:
   - along ρ, the sets of winning and losing players satisfy the original constraint (remember that we aim at solving the constrained existence problem);
   - any deviation along ρ leads to a state that is winning for A_1. In other terms, pick a state $h = (s, \text{Agt}, m_{\text{Agt}})$ of A_2 along ρ, and pick a successor $h' = (t, P)$ of h such that $t \neq \text{Tab}(s, m_{\text{Agt}})$. Then the algorithm checks that $t \in W(L \cap P)$.

The algorithm accepts the input M if it succeeds in finding the sets W and the path ρ such that all the checks are successful. This algorithm is in NP, and will be used as a second oracle.

We now show that if M is exactly the sum of the $w(L)$, then the algorithm accepts M if, and only if, there is a Nash equilibrium satisfying the constraint, i.e., if, and only if, A_1 has a winning strategy from (s, Agt) in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$.

First assume that the algorithm accepts M. This means that it is able, for each L, to find sets $W(L)$ of states whose complement does not intersect the winning region of $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$. Since M is assumed to be the exact sum of $w(L)$ and the size of the sets $W(L)$ sum up to M, we deduce that $W(L)$ is exactly the winning region of A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$. Now, since the algorithm accepts, it is also able to find a (lasso-shaped) path ρ only visiting states having Agt as the second component. This path has the additional property that any "deviation" from a state of player A_2 along this path ends up in a state that is winning for A_1 for players in $L \cap P$, where P is the set of suspects for the present deviation. This way, if during ρ, A_2 deviates to a state (t, P), then A_1 will have a strategy to ensure that along any subsequent play, the objectives of players in $L \cap P$ (in $\mathcal{G}$) are not fulfilled, so that along any run ρ', the players in $L \cap L(\rho')$ are losing for their objectives in $\mathcal{G}$, so that A_1 wins in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$.

Conversely, assume that there is a Nash equilibrium satisfying the constraint. Following Lemma 5, we assume that the outcome of the corresponding strategy profile has the form $\pi \cdot \tau^\omega$. From Lemma 28, there is a winning strategy for A_1 in $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$ whose outcome when A_2 obeys follows the outcome of the Nash equilibrium. As a consequence, the outcome when A_2 obeys is a path ρ that the algorithm can guess. Indeed, it must satisfy the constraints, and any deviation from ρ with set of suspects P ends in a state where A_1 wins for the winning condition of $\mathcal{H}(\mathcal{G}, \mathbf{v}(L))$, hence also for the winning condition of $\mathcal{H}(\mathcal{G}, \mathbf{v}(L \cap P))$, since any path ρ' visiting (t, P) has $L(\rho') \subseteq P$.

Finally, our global algorithm is as follows: we run the first oracle for all the states and all the sets L that are subsets of a set of suspects (we know that there are polynomially many such inputs). We also run the second algorithm on all the possible values for M, which are also polynomially many. Now, from the answers of the first oracle, we compute the exact value M, and return the value given by the second on that input. This algorithm runs in $\mathsf{P}_{\parallel}^{\mathsf{NP}}$. ☐

We now turn to the hardness proof:

Lemma 52. *For Büchi objectives with the parity preorder, the constrained existence problem is* $\mathsf{P}_{\parallel}^{\mathsf{NP}}$*-hard.*

Proof. The main reduction is an encoding of the $\mathsf{PARITY\ SAT}$ problem, where the aim is to decide whether the number of satisfiable instances among a set of formulas is even. This problem is known to be complete for $\mathsf{P}_{\parallel}^{\mathsf{NP}}$ [7].

Before tackling the whole reduction, we first develop some preliminaries on single instances of SAT, inspired from [5]. Let us consider an instance $\phi = C_1 \wedge \cdots \wedge C_n$ of SAT, where $C_i = \ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3}$, and $\ell_{i,j} \in \{x_k, \neg x_k \mid 1 \leq k \leq p\}$. With ϕ, we associate a three-player game $N(\phi)$, depicted on Fig. 12 (where the first state of $N(\phi)$ is controlled by A_1, and the first state of each $N'(C_j)$ is concurrently controlled by A_2 and A_3). For each variable x_j, players A_2 and A_3 have the following target sets:

$$T_{2j}^{A_2} = \{x_j\} \qquad T_{2j+1}^{A_2} = \{\neg x_j\} \qquad\qquad T_{2j+1}^{A_3} = \{x_j\} \qquad T_{2j}^{A_3} = \{\neg x_j\}$$

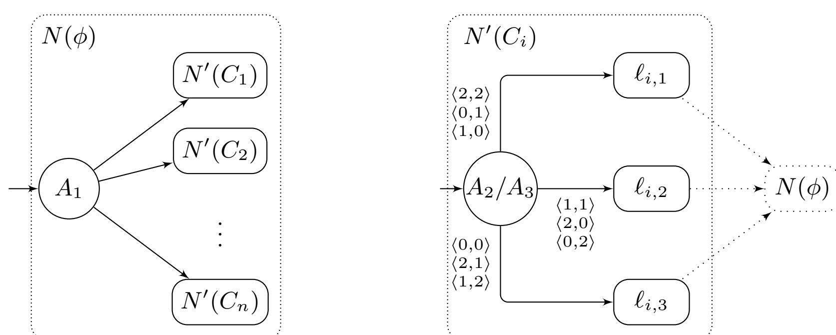


Fig. 12. The game $N(\phi)$ (left), where $N'(C_i)$ is the module on the right.

This construction enjoys interesting properties, given by the following lemma:

Lemma 53. *If the formula ϕ is not satisfiable, then there is a strategy for player A_1 in $N(\phi)$ such that players A_2 and A_3 lose. If the formula ϕ is satisfiable, then for any strategy profile σ_{Agt}, one of A_2 and A_3 can change her strategy and win.*

Proof. We begin with the first statement, assuming that ϕ is not satisfiable and defining the strategy for A_1. With a history h in $N(\phi)$, we associate a valuation $v^h\colon \{x_k \mid k \in [1,p]\} \to \{\top, \bot\}$ (where p is the number of distinct variables in ϕ), defined as follows:

$$v^h(x_k) = \top \;\Leftrightarrow\; \exists m.\ h_m = x_k \wedge \forall m' > m.\ h_{m'} \neq \neg x_k \qquad \text{for all } k \in [1,p]$$

We also define $v^h(\neg x_k) = \neg v^h(x_k)$. Under this definition, $v^h(x_k) = \top$ if the last occurrence of x_k or $\neg x_k$ along h was x_k. We then define a strategy σ_1 for player A_1: after a history h ending in an A_1-state, we require $\sigma_1(h)$ to go to $N'(C_i)$ for some C_i (with least index, say) that evaluates to false under v^h (such a C_i exists since ϕ is not satisfiable). This strategy enforces that if $h \cdot \sigma_1(h) \cdot \ell_{i,j}$ is a finite outcome of σ_1, then $v^h(\ell_{i,j}) = \bot$, because A_1 has selected a clause C_i whose literals all evaluate to $\bot$. Moreover, $v^{h\cdot\sigma_1(h)\cdot\ell_{i,j}}(\ell_{i,j}) = \top$, so that for each j, any outcome of σ_1 will either alternate between x_k and $\neg x_k$ (hence visit both of them infinitely often), or no longer visit any of them after some point. Hence both A_2 and A_3 lose.

We now prove the second statement. Let v be a valuation under which ϕ evaluates to true, and σ_{Agt} be a strategy profile. From σ_{A_2} and σ_{A_3}, we define two strategies σ'_{A_2} and σ'_{A_3}. Consider a finite history h ending in the first state of $N'(C_i)$, for some i. Pick a literal $\ell_{i,j}$ of C_i that is true under v (the one with least index, say). We set

$$\sigma'_{A_2}(h) = [j - \sigma_{A_3}(h) \pmod 3] \qquad \sigma'_{A_3}(h) = [j - \sigma_{A_2}(h) \pmod 3].$$

It is easily checked that, when σ_{A_2} and σ'_{A_3} (or σ'_{A_2} and σ_{A_3}) are played simultaneously in the first state of some $N'(C_i)$, then the game goes to $\ell_{i,j}$. Thus under those strategies, any visited literal evaluates to true under v, which means that at most one of x_k and $\neg x_k$ is visited (infinitely often). Hence one of A_2 and A_3 is winning, which proves our claim. $\square$

We now proceed by encoding an instance

$$\exists x_1^1, \dots x_k^1.\ \phi^1(x_1^1, \dots, x_k^1)$$
$$\dots$$
$$\exists x_1^m, \dots x_k^m.\ \phi^m(x_1^m, \dots, x_k^m)$$

of PARITY SAT into a game with Büchi objectives and the parity preorder. The game involves the three players A_1, A_2 and A_3 of the game $N(\phi)$ defined above, and it will contain a copy of $N(\phi^r)$ for each $1 \leq r \leq m$. The objectives of A_2 and A_3 are the unions of their objectives in each $N(\phi^r)$, e.g. $T_{2j}^{A_2} = \{x_j^1, x_j^2, \dots, x_j^m\}$.

For each such r, the game will also contain a copy of the game $M(\phi^r)$ depicted on Fig. 13. Each game $M(\phi^r)$ involves an extra set of players B_k^r, one for each variable x_k^r. Player B_k^r control the literal states $\ell_{i,j}^r$ when $\ell_{i,j}^r = \neg x_k^r$, then having the opportunity to go to state $\bot$. There is no transition to $\bot$ for literals of the form x^r_k. In $M(\phi^r)$, assuming that the players B^r_k will not play to $\perp$, then A_1 has a strategy that does not visit both x^r_k and $\neg x^r_k$ for every k if, and only if, formula ϕ^r is satisfiable. Finally, the objectives of B^r_k are $T_1^{B^r_k} = \{x^r_k\}$ and $T_0^{B^r_k}$ is the complement of $T_1^{B^r_k}$. In other terms, the aim of B^r_k is to visit x^r_k only a finite number of times. This way, in a Nash equilibrium, it cannot be the case that both x^r_k and $\neg x^r_k$ are visited infinitely often: it would imply that B^r_k loses but could improve her payoff by going to $\perp$ (actually, $\neg x^r_k$ should not be visited at all if x^r_k is visited infinitely often).

In order to test the parity of the number of satisfiable formulas, we then define two families of modules, depicted on Fig. 14 to 17. Finally, the whole game $\mathcal{G}$ is depicted on Fig. 18. In that game, the objective of A_1 is to visit infinitely often the initial state init.

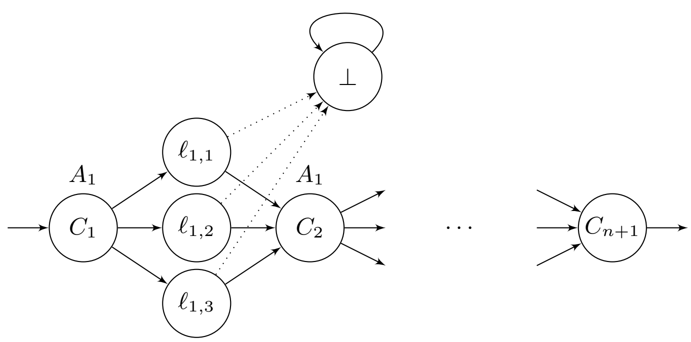


Fig. 13. Module $M(\phi)$, where $\phi = C_1 \wedge \cdots \wedge C_n$ and $C_i = \ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3}$

Lemma 54. *There is a Nash equilibrium in the game $\mathcal{G}$ where A_2 and A_3 lose and A_1 wins if, and only if, the number of satisfiable formulas is even.*

Proof. Assume that there is a Nash equilibrium in $\mathcal{G}$ where A_1 wins and both A_2 and A_3 lose. Let ρ be its outcome. As already noted, if ρ visits module $M(\phi^r)$ infinitely often, then it cannot be the case that both x^r_k and $\neg x^r_k$ are visited infinitely often in $M(\phi^r)$, as otherwise B^r_k would be losing and have the opportunity to improve her payoff. This implies that ϕ^r is satisfiable. Similarly, if ρ visits infinitely often the states of $H(\phi^r)$ or $G(\phi^r)$ that is controlled by A_2 and A_3, then it must be the case that ϕ^r is not satisfiable, since from Lemma 53 this would imply that A_2 or A_3 could deviate and improve her payoff by going to $N(\phi^r)$.

We now show by induction on r that if ρ goes infinitely often in module $G(\phi^r)$ then $\#\{j \leq r \mid \phi^r \text{ is satisfiable}\}$ is even, and that (if $n > 1$) this number is odd if ρ goes infinitely in module $H(\phi^r)$.

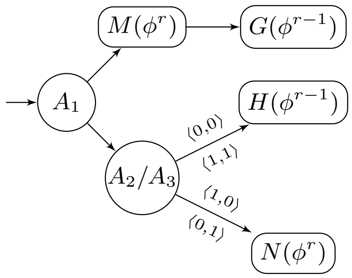


Fig. 14. Module $H(\phi^r)$ for $r \geq 2$

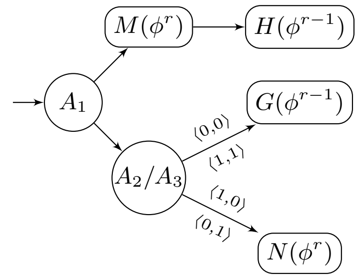


Fig. 15. Module $G(\phi^r)$ for $r \geq 2$

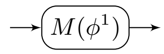


Fig. 16. Module $H(\phi^1)$

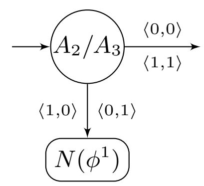


Fig. 17. Module $G(\phi^1)$

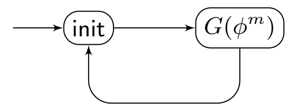


Fig. 18. The game $\mathcal{G}$

When $r = 1$, since $H(\phi^1)$ is $M(\phi^1)$, ϕ^1 is satisfiable, as noted above. Similarly, if ρ visits $G(\phi^1)$ infinitely often, it also visits its A_2/A_3-state infinitely often, so that ϕ^1 is not satisfiable. This proves the base case.

Assume that the result holds up to some $r-1$, and assume that ρ visits $G(\phi^r)$ infinitely often. Two cases may occur:

– it can be the case that $M(\phi^r)$ is visited infinitely often, as well as $H(\phi^{r-1})$. Then ϕ^r is satisfiable, and the number of satisfiable formulas with index less than or equal to $r-1$ is odd. Hence the number of satisfiable formulas with index less than or equal to r is even.
– it can also be the case that the state A_2/A_3 of $G(\phi^r)$ is visited infinitely often. Then ϕ^r is not satisfiable. Moreover, since A_1 wins, the play will also visit $G(\phi^{r-1})$ infinitely often, so that the number of satisfiable formulas with index less than or equal to r is even.

If ρ visits $H(\phi^r)$ infinitely often, using similar arguments we prove that the number of satisfiable formulas with index less than or equal to r is odd.

To conclude, since A_1 wins, the play visits $G(\phi^m)$ infinitely often, so that the total number of satisfiable formulas is even.

Conversely, assume that the number of satisfiable formulas is even. We build a strategy profile, which we prove is a Nash equilibrium in which A_1 wins and A_2 and A_3 lose. The strategy for A_1 in the initial states of $H(\phi^r)$ and $G(\phi^r)$ is to go to $M(\phi^r)$ when ϕ^r is satisfiable, and to state A_2/A_3 otherwise. In $M(\phi^r)$, the strategy is to play according to a valuation satisfying ϕ^r. In $N(\phi^r)$, it follows a strategy along which A_2 and A_3 lose (this exists according to Lemma 53). This defines the strategy for A_1. Then A_2 and A_3 are required to always play the same move, so that the play never goes to some $N(\phi^r)$. In $N(\phi^r)$, they can play any strategy (they lose anyway, whatever they do). Finally, the strategy of B_k^r never goes to $\perp$.

We now explain why this is the Nash equilibrium we are after. First, as A_1 plays according to fixed valuations for the variables x_k^r, either B_k^r wins or she does not have the opportunity to go to $\perp$. It remains to prove that A_1 wins, and that A_2 and A_3 lose and cannot improve (individually). To see this, notice that between two consecutive visits to init, exactly one of $G(\phi^r)$ and $H(\phi^r)$ is visited. More precisely, it can be observed that the strategy of A_1 enforces that $G(\phi^r)$ is visited if $\#\{r < r' \leq m \mid \phi^{r'} \text{ is satisfiable}\}$ is even, and that $H(\phi^r)$ is visited otherwise. Then if $H(\phi_1)$ is visited, the number of satisfiable formulas with index between 2 and m is odd, so that ϕ_1 is satisfiable and A_1 can return to init. If $G(\phi^1)$ is visited, an even number of formulas with index between 2 and m is satisfiable, and ϕ^1 is not. Hence A_1 has a strategy in $N(\phi^1)$ to make A_2 and A_3 lose, so that A_2 and A_3 cannot improve their payoffs. □

□

□

References

1. R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *J. ACM*, 49(5):672–713, 2002.
2. P. Bouyer, R. Brenguier, and N. Markey. Nash equilibria for reachability objectives in multi-player timed games. In CONCUR'10, LNCS 6269, p. 192–206. Springer, 2010.
3. P. Bouyer, R. Brenguier, N. Markey, and M. Ummels. Nash equilibria in concurrent games with Büchi objectives. In FSTTCS'11, LIPIcs, Mumbai, India, 2011. Leibniz-Zentrum für Informatik. To appear.
4. K. Chatterjee and M. Henzinger. An $O(n^2)$ time algorithm for alternating Büchi games. Technical Report IST-2011-0009, IST, Kleusterneuburg, Austria, 2011.
5. K. Chatterjee, T. A. Henzinger, and N. Piterman. Generalized parity games. In FoSSaCS'07, LNCS 4423, p. 153–167. Springer, 2007.
6. K. Chatterjee, R. Majumdar, and M. Jurdziński. On Nash equilibria in stochastic games. In CSL'04, LNCS 3210, p. 26–40. Springer, 2004.
7. G. Gottlob. NP trees and Carnap's modal logic. *Journal of the ACM*, 42(2):421–457, 1995.
8. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games*, LNCS 2500. Springer, 2002.
9. T. A. Henzinger. Games in system design and verification. In TARK'05, p. 1–4, 2005.
10. P. Hunter and A. Dawar. Complexity bounds for regular games. In MFCS'05, LNCS 3618, p. 495–506. Springer, 2005.
11. M. Jurdzinski. Deciding the winner in parity games is in $\mathsf{UP} \cap \mathsf{co\text{-}UP}$. *Inf. Process. Lett.*, 68(3):119–124, 1998.
12. F. Laroussinie, N. Markey, and G. Oreiby. On the expressiveness and complexity of ATL. *Logicical Methods in Computer Science*, 4(2), 2008.
13. D. A. Martin. Borel determinacy. *The Annals of Mathematics*, 102(2):363–371, 1975.

14. R. McNaughton. Infinite games played on finite graphs. *Annals of Pure and Applied Logic*, 65(2):149–184, 1993.
15. J. F. Nash, Jr. Equilibrium points in n-person games. *Proc. National Academy of Sciences of the USA*, 36(1):48–49, 1950.
16. J. Neumann, A. Szepietowski, and I. Walukiewicz. Complexity of weak acceptance conditions in tree automata. *Inf. Process. Lett.*, 84(4):181–187, 2002.
17. S. Paul and S. Simon. Nash equilibrium in generalised Muller games. In FSTTCS'09, LIPIcs 4, p. 335–346. LZI, 2010.
18. L. S. Shapley. Stochastic games. *Proc. National Academy of Sciences of the USA*, 39:1095–1100, 1953.
19. W. Thomas. Infinite games and verification (extended abstract of a tutorial). In CAV'02, LNCS 2404, p. 58–64. Springer, 2002.
20. M. Ummels. The complexity of Nash equilibria in infinite multiplayer games. In FoSSaCS'08, LNCS 4962, p. 20–34. Springer, 2008.
21. M. Ummels and D. Wojtczak. The complexity of Nash equilibria in limit-average games. In CONCUR'11, LNCS 6901, p. 482–496. Springer, 2011.